Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

Disseny i prototipat de solucions auto-adaptatives emprant arquitectures basades en microserveis. Una aplicació industrial pràctica.

Treball Fi de Màster

Máster Universitari en Enginyeria Informàtica

Autor: Climent Penadés, José

Tutor: Fons i Cors, Joan

Curs: 2019/2020

Resum

Les arquitectures basades en microserveis (µserveis) ofereixen un enfoc modern per al desenvolupament de solucions de programari que operen de manera natural distribuïdes en la xarxa, i que sorgeixen amb l'objectiu d'oferir una flexibilitat computacional alta. Per a això, aquest tipus d'infraestructures proveeixen de mecanismes per a dissenyar i desplegar solucions que puguen replicar-se (clonar-se, duplicar-se) o replegar-se (contraure's) dinàmicament, adaptant-se en funció de la càrrega de treball o "càrrega computacional" que tinguen a cada moment.

No obstant això, aquest tipus de solucions només tenen en compte aspectes d'adaptació en funció d'aquesta càrrega computacional dinàmica. Existeixen dominis (per exemple, el de Internet de les Coses), també dinàmics, on l'aplicació d'aquest tipus d'infraestructures podria proveir un entorn de computació adaptable a aspectes o requisits funcionals i operatius. En l'actualitat, no és habitual dissenyar i desenvolupar programari amb capacitats de computació autònoma, és a dir, capaços d'adaptar-se per a funcionar de manera òptima en funció del seu context operatiu.

En aquest projecte es pretenen integrar, des d'un punt de vista pràctic, tècniques que provenen de la teoria de control, utilitzant bucles de control per a incorporar capacitats d'adaptació als sistemes. En concret, s'utilitzarà l'aproximació FADA (desenvolupada pel grup Tatami del Centre d'investigació PROS) que proposa un enfocament pràctic per a desenvolupar programari autoadaptatiu aplicant els conceptes de la computació autònoma a través de bucles de control MAPE-K.

S'abordarà el disseny d'una xicoteta solució basada en µserveis que puga canviar la seua configuració operativa i de desplegament, de manera dinàmica. Ens centrarem en identificar situacions que requerisquen canvis en una infraestructura de µserveis i definirem operacions de canvi de la configuració de la solució a través de canvis en el propi desplegament de µserveis.

S'obtindrà el disseny basat en µserveis d'una solució basada en el domini real de la fàbrica Embalpack, on les capacitats de computació autònoma i d'autoadaptació siguen part del problema. Finalment, s'obtindrà un prototip funcional basat en l'aproximació FADA que done resposta al problema proposat en la mesura que siga possible.

Paraules clau: Arquitectures de µserveis, Computació Autònoma, Bucles de Control, MAPEK-K.

Resumen

Las arquitecturas basadas en microservicios (µservicios) son un enfoque moderno para el desarrollo de soluciones software que operan de manera natural distribuídas en la red, y que surgen con el objetivo de ofrecer una flexibilidad computacional alta. Para ello, este tipo de infraestructuras proveen de mecanismos para diseñar y desplegar soluciones que puedan replicarse (clonarse, duplicarse) o replegarse (contraerse) dinámicamente, adaptándose en función de la carga de trabajo o "carga computacional" en cada momento.

Sin embargo, este tipo de soluciones sólo tienen en cuenta aspectos de adaptación en función de esta carga computacional dinámica. Existen dominios (por ejemplo, en el de la Internet de las Cosas), también dinámicos, donde la aplicación de este tipo de infraestructuras podría proveer un entorno de computación adaptable a aspectos o requisitos funcionales y operativos. En la actualidad, no es habitual diseñar y desarrollar software con capacidades de computación autónoma, es decir, capaces de adaptar-se para funcionar de manera óptima en función de su contexto operativo.

En este proyecto se pretenden integrar, desde un punto de vista práctico, técnicas que provienen de la teoría de control, utilizando bucles de control para incorporar capacidades de adaptación a los sistemas. En concreto, se usará la aproximación FADA (desarrollada por el grupo Tatami del Centro de investigación PROS) que propone un enfoque práctico para desarrollar software autoadaptativo aplicando los conceptos de la computación autónoma a través de bucles de control MAPE-K.

Se abordará el diseño de una pequeña solución basada en µservicios que pueda cambiar su configuración operativa y de despliegue, de manera dinámica. Nos centraremos en identificar situaciones que requieran cambios en una infraestructura de µservicios y definiremos operaciones de cambio de configuración de la solución a través de cambios en el propio despliegue de µservicios.

Se obtendrá el diseño basado en µservicios de una solución basada en el dominio real de la fábrica Embalpack, donde las capacidades de computación autónoma y de autoadaptación sean parte del problema. Se obtendrá, además, un prototipo funcional basado en la aproximación FADA que dé respuesta al problema propuesto en la medida de lo posible.

Palabras clave: Arquitecturas de µservicios, Computación Autónoma, Bucles de Control, MAPEK-K.

Abstract

Microservices-based architectures (µservices) are a modern approach to developing software solutions that operate naturally distributed on the network, and that emerge with the aim of offering high computational flexibility. To do this, this type of infrastructure provides mechanisms to design and deploy solutions that can be replicated (cloned, duplicated) or retracted (contracted) dynamically, adapting according to the workload or "computational load" at each moment.

However, these types of solutions only consider adaptation aspects based on this dynamic computational load. There are domains (for example, in the Internet of Things), also dynamic, where the application of this type of infrastructure could provide a computing environment adaptable to functional and operational aspects or requirements. Currently, it is unusual to design and develop software with autonomous computing capabilities and able to adapt to function optimally based on your operating context.

In this project, the intention is to integrate, from a practical point of view, techniques that come from the theory of control, using control loops to incorporate adaptation capabilities to the systems. Specifically, the FADA approach (developed by the Tatami group of the PROS Research Centre) will be used, which proposes a practical approach to develop self-adaptive software applying the concepts of autonomous computing through MAPE-K control loops

The design of a small µservice-based solution that can dynamically change its operational and deployment configuration will be addressed. We will focus on identifying situations that require changes in a µservice infrastructure and define operations to change the configuration of the solution through changes in the deployment of µservices.

We will obtain the design based on µservices of a solution based on the real domain of the Embalpack factory, where the autonomous computing and self-adaptation capabilities are part of the problem. In addition, a functional prototype will be obtained based on the FADA method that responds to the proposed problem as much as possible.

Keywords: µservice architectures, Autonomous Computing, Control Loops, MAPEK-K.

Taula de continguts

Taula de figures

1. Introducció

Avui en dia estan començant a sorgir molts sistemes basats en µserveis, ja que, tot i tindre cert nivell de dificultat i complicació a l'hora de desenvolupar una solució d'aquest tipus, (generalment, els sistemes monolítics o tradicionals són més senzills de desenvolupar) solen tindre majors beneficis que els sistemes tradicionals, ja que aquests poden adaptar-se a la capacitat de còmput disponible/necessària dinàmicament, enfront dels sistemes monolítics que no poden adaptar-se de manera tan òptima com els µserveis.

A més, aquest tipus de solucions solen perdurar més en el temps però, i si fórem capaços d'augmentar l'eficàcia d'aquests sistemes? En aquest treball es plantejarà la possibilitat d'afegir computació autònoma a una solució basada en µserveis de manera que, a més de les qualitats que posseeix aquest tipus de solucions, el sistema siga capaç d'optimitzar-se i d'adaptar-se depenent del context en què treballem.

Per tant, una vegada coneguda la premissa, s'intentarà, a partir d'una situació real d'una empresa (Embalpack) adaptar el sistema basat en µserveis actual, afegint-li bucles de control per poder gestionar la solució i fer que s'optimitze en funció del context actual de l'empresa.

Una vegada analitzada la situació, es procedirà a realitzar la implementació de la nova solució i proposar una sèrie d'escenaris per veure com es comporta el nou sistema. Per poder realitzar l'anàlisi del nou sistema, s'implementarà un prototip funcional on es provaran diversos escenaris per observar com es comportaria el sistema i veure quines millores aporta el nou sistema respecte a l'anterior.

1.1. Motivació

A l'actualitat estan començant a sorgir molts sistemes basats en µserveis, per tant, en un entorn industrial o empresarial es poden trobar sistemes tradicionals, on la solució que utilitzen es va quedant obsoleta o curta de capacitats i solucions basades en µserveis que perseveren millor amb el pas del temps.

Per altra banda la computació autònoma permet incorporar noves capacitats al programari que el fan ser més conscient d'ell mateix i de l'entorn operatiu en el que funciona i permet desplegar-li coneixement per a que siga capaç d'autogestionar-se.

Aquestes noves capacitats, que no es solen tenir en compte a l'hora de fer l'enginyeria dels sistemes, permet fer els sistemes més resilients i a la vegada, menys dependents del factor humà per a evitar o corregir errades.

La motivació principal d'aquest projecte és fer una recerca per veure com seria desenvolupar una solució emprant tècniques d'ambdós mons, és a dir, aconseguir un sistema basat en µserveis amb capacitats de computació autònoma de manera que es puga adaptar a les necessitats depenent del context, l'entorn i els recursos disponibles. A més, es pretén analitzar si aquesta proposta és beneficiosa i observar quin impacte podria tindre en la indústria l'ús de les dos tècniques (µserveis i capacitat de còmput autònoma) en conjunt.

1.2. Objectius

Aquest projecte té com a objectius principals comprovar si es poden **fusionar dues tècniques innovadores** i **veure l'impacte i els beneficis que tindria l'ús d'aquestes tècniques en conjunt**. Per arribar a aquest objectiu es dotarà al sistema inicial (desenvolupat mitjançant µserveis) de computació autònoma, mitjançant bucles de control MAPE-K. Per a obtenir aquesta capacitat s'estudiaran i desenvoluparan diversos patrons de disseny de µserveis i diverses estratègies per veure si es poden d'integrar al bucle de control i així dotar al sistema de computació autònoma.

Els objectius secundaris d'aquest objectiu principal són:

- Estudi de patrons de desenvolupament.
- Desenvolupament de prototips funcionals.
- Realització de proves d'adaptabilitat del sistema desenvolupat.
- Recaptació de resultats.

1.3. Impacte esperat

S'espera obtindre un resultat satisfactori, on finalment s'aconsegueix demostrar que és factible l'ús d'ambdues tècniques, de manera que els µserveis adquirisquen capacitats de computació autònoma. Per fer açò, primerament es farà un estudi conceptual de manera que s'estudiaran els possibles reptes i problemes d'aquest tipus de solucions i així, poder analitzar si una solució d'aquest tipus és viable.

Si aquesta premissa funciona conceptualment, s'espera poder abordar-ho a nivell industrial, sabent amb anterioritat el tipus de problemes, reptes i eines de les que es disposa per a poder afrontar el disseny i desenvolupament d'aquest tipus de sistemes.

1.4. Metodologia i Pla de treball

Aquest treball és, en la seua major part, teòric. Per tant, li hem dedicat un treball previ d'uns tres mesos a acordar la temàtica concreta, fer treball de recerca sobre el tema escollit i dissenyar l'experiment. Una vegada concretada la temàtica, ens centrarem en desenvolupar les funcionalitats necessàries per a dur endavant el projecte i poder realitzar un prototip basat en un escenari industrial per veure quin impacte produiria aquest tipus de solucions. Finalment realitzarem les proves pertinents i documentarem els resultats.

Figura 1: Flux de treball.

El primer que hem de fer per a la realització d'aquest treball és una etapa de recerca, on descobrirem les tecnologies, les seues característiques i com es podrien combinar. Durant aquesta etapa hem de veure quines tècniques o característiques podem utilitzar d'una tecnologia per a integrar-les en l'altra.

Una vegada hem fet l'anàlisi prèvia practicarem el que hem aprés en un escenari generalitzat de manera que no ens hem de preocupar de cap qualitat especial que puga tindre un sistema en concret. A partir d'aquest escenari general, posarem en pràctica el que hem aprés i intentarem aplicar diverses estratègies per integrar ambdues tecnologies.

Aleshores, si hem aconseguit fer la integració adequadament i conceptualment té sentit el que hem desenvolupat, transferirem els coneixements a un entorn industrial, on desenvoluparem un prototip i realitzarem proves per veure si el que conceptualment funcionava, també ho fa en un escenari industrial. Per aconseguir açò haurem de presentar uns escenaris on tindre en compte diversos tipus de situacions i veure com es desenvolupa el sistema per a resoldre-les.

Finalment, ens quedarà analitzar els resultats obtinguts i veure si realment l'experiment ha resultat ser exitós o, per altra banda, no hem obtingut el resultat que esperàvem i, per tant, aquesta tècnica no ens permet obtindre una solució de programari viable.

1.5. Estructura

Aquest projecte es divideix en set capítols, bibliografia i un annex. En aquesta secció ens disposem a explicar de manera breu el contingut de cada secció i el treball realitzat dins de la mateixa.

- **Capítol 1 – Introducció:** En aquest capítol es comenta el propòsit del treball, és a dir, com s'ha decidit el projecte a realitzar, els objectius que cal dur a terme, la metodologia utilitzada per al desenvolupament d'aquest i l'impacte esperat una vegada s'ha finalitzat la seua implementació.

- **Capítol 2 – Estat de l'art:** Explicació de l'estat actual de les metodologies i tècniques utilitzades per a la realització d'aquest treball.

- **Capítol 3 – Enginyeria de µserveis autònoms:** S'analitza el problema que volem afrontar en aquest projecte i es descriuen les tècniques i mètodes utilitzats per a afrontar-lo.

- **Capítol 4 – Disseny de µserveis autònoms:** Disseny de la solució per al problema plantejat a partir dels mètodes descrits anteriorment.

- **Capítol 5 – Implementació de µserveis autònoms:** Desenvolupament dels diferents components que conformen la solució d'acord al disseny que s'ha obtingut després de fer l'anàlisi.

- **Capítol 6 – Aplicació industrial pràctica:** S'explicarà quines característiques són les que posseeix l'escenari industrial escollit i com es desenvoluparà i es comportarà el sistema industrial al aplicar-li les tècniques de desenvolupament del capítol anterior.

- **Capítol 7 – Conclusions:** Conclusions del projecte, es resoldrà si s'han aconseguit els objectius que s'han plantejat al projecte. S'analitzarà com es relaciona aquest projecte amb els estudis cursats durant el màster. S'estudiaran millores que es podrien fer a l'aplicació, resultats esperats després d'aplicar les millores i futurs projectes relacionats amb aquest.

- **Bibliografia:** Fonts d'informació d'on hem obtingut les dades per a la realització d'aquest projecte.

- **Annex:** Característiques dels components de la solució industrial i seqüències d'execució dels escenaris de prova.

2. Estat de l'art

Per a aquest treball hi ha dues aproximacions principals que són de rellevància: la computació autònoma i l'aplicació d'arquitectures basades en µserveis. En les següents seccions es presenten ambdues aproximacions, i es conclou amb una anàlisi crítica sobre les aportacions i mancances de cadascuna d'elles.

2.1. Arquitectura Orientada a Serveis

L'arquitectura orientada a serveis (SOA) és un estàndard que presenta tots els processos de negoci d'una manera orientada a serveis. Les dependències entre serveis, com ara serveis web, actius de serveis EIS (*Enterprise Information System*), fluxos de treball i bases de dades es minimitzen i s'oculta la implementació de qualsevol servei [1].

Aquesta arquitectura consta d'una col·lecció de serveis autònoms i petits. Els serveis són independents entre si i cada un ha d'implementar una funcionalitat individual. Per aquest motiu, es pot considerar com un enfocament per a desenvolupar una sola aplicació com un conjunt de petits serveis, cadascun funcionant al seu propi procés i comunicant-se amb l'objectiu d'aconseguir realitzar una funció concreta amb la mateixa efectivitat que una aplicació tradicional [2].

Al ser components amb funcionalitats específiques, aquest tipus de sistemes són altament escalables ja que podem replicar el component que siga necessari en qualsevol moment sense necessitat de replicar tota la solució. També, a causa de les característiques úniques d'aquest tipus de sistemes, proporcionen una forma ben definida d'exposició i invocació de serveis, la qual cosa facilita la interacció entre diferents sistemes propis o de tercers.

L'objectiu de l'arquitectura orientada a serveis és separar la lògica d'integració de negoci de la implementació. Algunes característiques d'aquesta metodologia són:

- Al ser serveis independents, cadascun d'aquests pot estar desenvolupat seguint una arquitectura o utilitzar una tecnologia totalment diferent de la resta.
- Cada servei exposa o requereix de certes dades, per tant, cada servei proporciona o rep la informació d'una manera concreta.

- Faciliten l'intercanvi d'informació, ja que fent una petició al servei desitjat pots rebre la informació que aquest proporciona, sense necessitat de canviar de sistema d'intercanvi d'informacions.
- Aquest sistema és altament escalable, ja que podem replicar els serveis segons el que necessitem.
- Permet reaccionar amb agilitat davant els canvis de les condicions de les empreses i beneficiar-se dels mateixos.
- Programari sostenible en el temps, al poder canviar qualsevol servei sense necessitat de canviar la solució completa, aquest tipus de solucions tenen més esperança de vida que les tradicionals o monolítiques.

Quan un desenvolupador va a fer una aplicació amb serveis seguint aquest patró de disseny, es fixa en què la seua solució complisca aquests requisits [3, 4]:

- Els components s'acoblen de forma oberta. Un component que accedeix a un altre no necessita conèixer les estructures de dades, les crides a altres components, la gestió de transaccions, etc. de l'altre component.
- Els components són configurables. S'ha de parèixer a un diagrama de configuració. Els components s'han de poder afegir, suprimir i configurar-se amb diferents procediments per crear aplicacions noves.
- Els components poden treballar conjuntament. Qualsevol component pot treballar amb un altre, inclosos els components creats per entorns de desenvolupament de proveïdors diferents.
- Els components són independents de la ubicació.

Com a conseqüència de les característiques que presenta aquesta arquitectura podem dir que SOA estarà formada per la combinació de tecnologies, productes, API's (Interfícies relacionals d'aplicacions) extensions de la tecnologia de suport i altres parts diverses. Per tant, aquests principis de disseny creen una arquitectura flexible amb capacitat d'adaptació segons les necessitats de còmput.

2.1.1. Arquitectura Orientada a (micro)Serveis vs Arquitectura Monolítica

Per veure els avantatges i els inconvenients que posseeix aquest tipus d'arquitectura utilitzarem l'arquitectura tradicional o monolítica per fer una comparativa i veure els beneficis que aporta aquesta nova arquitectura a diferència de la tradicional.

Primerament definirem el concepte de µservei. Els µserveis són petits i independents, i estan acoblats de forma imprecisa. Açò implica que un µservei por tindre unes característiques, tenir un sistema d'emmagatzematge o tractament de dades i estar desenvolupat en un llenguatge o baix una arquitectura única. A causa d'aquestes característiques, per comunicar-se entre si, solen emprar API's per garantir que la comunicació siga efectiva [3].

Per tant, es pot afirmar que l'arquitectura monolítica es basa en un gran programa que realitza totes les funcions mentre que l'arquitectura orientada a µserveis es basa en una gran quantitat de serveis, cadascun oferint una funció diferent i cooperant per oferir una solució completa [5].

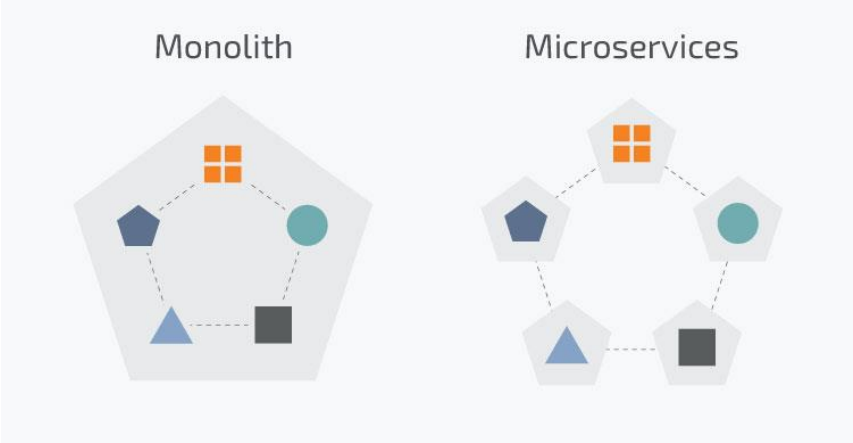

Figura 2: Arquitectura Monolítica vs Arquitectura Orientada a µserveis.

Una vegada està clar el concepte de µservei, es passarà a comentar els avantatges i inconvenients d'aquesta arquitectura [6].

El patró d'Arquitectura de µserveis té una sèrie d'avantatges importants [3]. En primer lloc, tracta el problema de la complexitat i l'adaptació a les noves tecnologies. Descomposa el que d'una altra manera seria una aplicació monolítica enorme en un conjunt de serveis. Si bé la quantitat total de funcionalitats és la mateixa, l'aplicació s'ha desglossat en un conjunt de serveis assequibles on cadascun s'ocupa d'una única funcionalitat.

El patró d'Arquitectura de µserveis obliga a un nivell de modularitat que a la pràctica és extremadament difícil d'aconseguir amb el codi base d'una típica aplicació monolítica. En conseqüència, els serveis individuals són molt més ràpids per desenvolupar-se, i molt més fàcils d'entendre i mantenir.

Aquesta arquitectura també permet que cada servei puga ser desenvolupat depenent de les necessitats de cada desenvolupador, amb el llenguatge, tècniques, mètodes, etc. que ell considere oportunes ja que, independentment de la forma en què es desenvolupe aquest s'hauria de poder comunicar mitjançant una API. Aquesta llibertat fa que els desenvolupadors ja no estiguen obligats a utilitzar les tecnologies, possiblement obsoletes, que existien a l'inici d'un nou projecte i puguen començar a desenvolupar el nou servei amb tecnologies més actuals. A més, com que els serveis són relativament petits, es fa possible reescriure un servei antic per actualitzar-lo a una tecnologia actual.

Els µserveis es poden desplegar de manera independent. Els desenvolupadors no han de coordinar mai el desplegament de canvis locals al seu servei. Aquest tipus de canvis es poden implementar tan prompte com s'hagen provat.

A més, com ja hem comentat abans, el patró d'Arquitectura de µserveis permet escalar de forma independent cada servei. Podem implementar només el nombre d'instàncies de cada servei que satisfan les seues restriccions de capacitat i disponibilitat. A més, podem utilitzar els recursos que s'adapten millor als requisits de recursos d'un servei.

Com qualsevol tecnologia, totes presenten avantatges i inconvenients i aquesta no anava a ser l'excepció. Un inconvenient important dels µserveis és la complexitat que es desprèn del fet que una aplicació de µserveis és un sistema distribuït, entre que les aplicacions monolítiques que no necessiten d'aquest sistema de comunicacions, ja que al ser una sola aplicació, les comunicacions d'aquest tipus no són necessàries i per tant és més fàcil de desenvolupar.

Els desenvolupadors han de triar i implementar un mecanisme de comunicació entre processos basat en missatgeria. A més, també han d'escriure codi per gestionar un error parcial ja que la destinació d'una sol·licitud pot ser lenta o no disponible. Per tant el sistema de comunicacions ha de ser robust per garantir la consistència de les dades i, en conseqüència, el seu desenvolupament és més complex que el d'una aplicació monolítica.

Un altre desafiament amb µserveis és l'arquitectura de bases de dades particionada davant la base de dades única del sistema tradicional monolític. En moltes solucions de tipus empresarial la fluctuació de dades és freqüent i es necessita enregistrar dades, modificar-les o eliminar-les de la millor manera possible. En una aplicació monolítica sols tenim una base de dades mentre que en una aplicació amb µserveis, cada µservei controla les seues dades i les emmagatzema a una base de dades pròpia. Aquest fet augmenta la dificultat a l'hora de fer el desenvolupament dels µserveis i el seu tractament de dades.

També és molt més complex provar una aplicació de µserveis. Per exemple, amb un marc modern com "Spring Boot" és banal escriure una classe de prova que inicie una aplicació web monolítica i pose a prova la seua API REST. En canvi, una classe de prova similar per a un servei hauria de llançar aquest servei i tots els serveis que en depenen. A més, per fer actualitzacions i les seues possibles proves, cal tindre en compte, a banda d'allò anteriorment dit, les dependències i l'ordre de llançament dels serveis per a que no es produïsquen conflictes [7].

Una aplicació monolítica només es desplega en un conjunt de servidors idèntics darrere d'un equilibrador de càrrega tradicional. Cada instància d'aplicació es configura amb les ubicacions (*host* i ports) dels serveis d'infraestructura, com ara la base de dades i un agent de missatges. En canvi, una aplicació de µserveis consisteix generalment en un gran nombre de serveis. Cada servei tindrà diverses instàncies d'execució. Es tracta de moltes parts mòbils que cal configurar, desplegar, escalar i controlar.

A més, µservei són dinàmics i es poden desplegar en qualsevol sistema, canviar d'ubicació a voluntat, replegar-se o desplegar-se'n de nous així que cal tindre en compte que hem de controlar la ubicació i la disponibilitat en tot moment [8]. Per tant, també serà necessari implementar un mecanisme de descobriment de serveis que permeta a un servei descobrir les ubicacions (*host* i ports) de qualsevol altre servei amb el qual necessite comunicar-se. En conseqüència, desplegar amb èxit una aplicació de µserveis requereix un major control dels mètodes de desplegament per part dels desenvolupadors i un alt nivell d'automatització.

Una vegada enumerats els diferents avantatges i inconvenients del sistema de µserveis davant el sistema monolític, s'aportaran una sèrie de raons per les quals s'ha decidit desenvolupar un sistema basat en µserveis i no a un sistema tradicional o monolític.

S'ha decidit emprar un sistema basat en µserveis ja que aquest perdura millor en el temps, s'obté un sistema més escalable i amb millor adaptació a la infraestructura de què disposem. A més, l'objectiu principal d'aquest projecte és demostrar que és possible combinar aquest tipus d'arquitectures amb computació autònoma i veure quins beneficis pot aportar la combinació d'aquestes tècniques.

2.1.2. Patrons de disseny d'arquitectura de µserveis.

A continuació, es mostraran algunes de les tècniques o patrons que es poden utilitzar per a dissenyar una solució basada en µserveis. Aquests seran alguns dels que s'utilitzaran en un futur per a dissenyar les polítiques del bucle de control i que utilitzarà per a fer les adaptacions corresponents en cas de necessitar-ho [6, 8].

- ***API-Gateway*:** Aquest patró permet accedir a múltiples serveis sense necessitat de conéixer el destinatari, és a dir, fent sols una crida al servei que proporciona *l'API-Gateway*, aquest s'encarrega de redireccionar la petició al servei que corresponga. El que fa aquest patró és col·locar un intermediari entre aquells µserveis que cridarem fent ús de l'api i l'usuari final. Una vegada col·locat, les crides a tots aquests serveis es faran per l*'API-Gateway* i per tant sols farà falta una crida i no la gran quantitat de crides que farien falta sense aquest sistema (una per servei). L'*API-Gateway* encapsula l'arquitectura del sistema intern i proporciona una API adaptada a cada client. Pot ser que tinga altres responsabilitats com ara autenticació, supervisió, equilibri de càrregues, memòria cau, gestió i conformació de sol·licituds i maneig de respostes estàtiques. L'*API-Gateway* és responsable de l'encaminament de la sol·licitud, la composició i la traducció de protocols. Totes les sol·licituds dels clients passen primer per la passarel·la de l'API. A continuació, dirigeix les sol·licituds al µservei adequat. L'*API-Gateway* tractarà sovint una sol·licitud invocant múltiples µserveis i agregant els resultats.

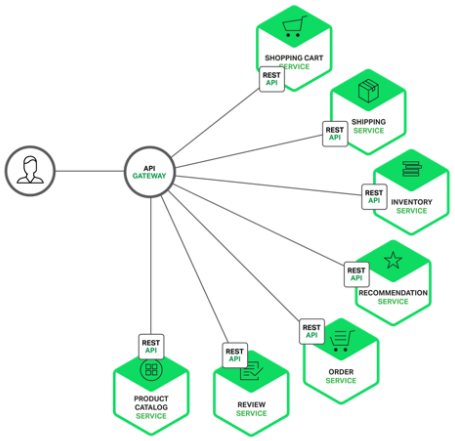

Figura 3: *API-Gateway*.

- ***Load Balancer*:** Quan un µservei està suportant una gran càrrega computacional a causa del seu ús elevat és possible que es genere un coll de botella ja que es generen més peticions de les que pot respondre. En aquest instant, és quan s'aplica aquest patró. El que fem és col·locar un intermediari que gestione les peticions que arriben i a més, despleguem diverses instàncies d'aquest µservei de manera que així l'intermediari pot repartir les peticions entre les instàncies d'un mateix µservei, cosa que permet que les peticions que abans generaven un coll de botella es pugen resoldre més ràpidament i així s'elimina el coll de botella que s'havia produït.

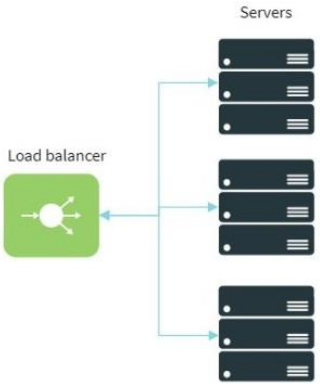

Figura 4: *Load Balancer*.

- ***Circuit Breaker*:** Aquest patró ens permet simular que un µservei funciona correctament, és a dir, ens protegeix davant les possibles errades o desconnexions del servei de manera que si aquest deixa de funcionar correctament, el *Circuit Breaker* que conté una memòria cau de l'últim estat correcte del µservei es connecta i simula que aquest continua funcionant correctament. Aquest patró funciona col·locant un intermediari entre un µservei i l'altre de manera que l'intermediari monitoritza el µservei, si arriba el moment en què aquest comença a fallar i supera el límit d'errades, el *Circuit Breaker* el desconnecta i el substitueix temporalment utilitzant la memòria cau emmagatzemada. Una vegada el servei torna a funcionar correctament, el *Circuit Breaker* reenganxa el servei i es desconnecta sense que l'usuari que està generant les peticions se n'adone que hi ha hagut un moment en què ha deixat de funcionar correctament.

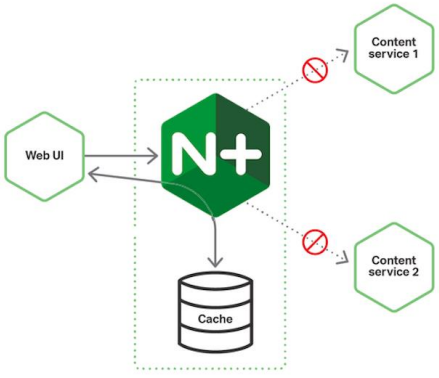

Figura 5: *Circuit Breaker*.

- **Patrons de desplegament:** Depenent dels recursos que podem tindre en un determinat moment ens interessarà més una estratègia o una altra, les més comunes són:

    - ***Service per container*:** Aquesta estratègia de desplegament de µserveis garanteix que un contenidor sols contindrà un µservei. El que fem a aquest sistema és empacar cada µservei com a imatge de contenidor de manera que al desplegar els serveis es despleguen dins d'un contenidor propi garantint que quan es desplega el contenidor sols contindrà un µservei.

    - ***Multiple instance per host*:** Aquest patró proporciona un o més amfitrions físics o virtuals i executa diverses instàncies de servei en cadascun. En molts aspectes, aquest és l'enfocament tradicional del desplegament d'aplicacions. Cada instància de servei s'executa en un port conegut d'un o més amfitrions.

    - ***Service per Virtual Machine*:** Quan s'utilitza aquest patró cada servei es col·loca com a una màquina virtual, de manera que el servei es llança mitjançant la imatge la màquina virtual en qüestió. Amb aquest sistema podem assignar els recursos que té disponibles cada instància de manera que no pot agafar recursos pertanyents a una altra instància.

- **Patró de tractament de dades:** Aquest patró es basa en escollir una estratègia d'emmagatzematge de dades depenent del nivell de consistència de dades o la rapidesa que requereix el nostre sistema. Els més utilitzats són aquests:

    - ***Database per Service*:** Aquesta estratègia d'emmagatzematge de dades es basa en què cada µservei guarda les dades a una base de dades pròpia, per tant si un µservei vol guardar alguna dada a la base de dades d'un altre ha de donar-li les dades al µservei en qüestió mitjançant un *broker* de missatgeria per a que aquest faça l'operació. D'aquesta manera podem garantir fins a cert nivell la consistència de les dades emmagatzemades.

- ***Shared Database*:** Aquesta estratègia d'emmagatzematge de dades es basa en què diversos µserveis guarden les dades a la mateixa base (generalment anomenada *Event Database*). Cada µservei té accés a la seua base de dades local i a la base compartida de manera que a la compartida s'escriuen les peticions de modificació i el µservei en qüestió llegeix aquesta taula i aplica els canvis pertinents a la seua base de dades local.

- ***Service Discovery*:** Aquest patró es basa en escollir l'estratègia de descobriment de serveis ja que aquests, es troben en constant moviment per tant les direccions IP són dinàmiques ja que poden canviar de màquina, replicar-se, replegar-se, desplegar-se... Els més comuns són:

  - ***Client-Side Discovery*:** Quan s'utilitza el *client-side discovery*, el client és responsable de determinar les ubicacions de xarxa de les instàncies de servei disponibles i les sol·licituds d'accés (en cas que es trobe sota un *Load Balancer*) a través d'elles. El client consulta un registre de serveis, que és una base de dades d'instàncies de servei disponibles. Aleshores, el client utilitza un algorisme de balanceig de càrrega per seleccionar una de les instàncies de servei disponibles i fa una sol·licitud. La ubicació de xarxa d'una instància de servei es registra al servei de registre quan s'inicia. S'elimina del registre de serveis quan finalitza la instància. El servei registrador d'instàncies s'acostuma a actualitzar periòdicament mitjançant un mecanisme de *Hearth-beat*.

  - ***Server-Side discovery*:** El client realitza una sol·licitud a un servei mitjançant un balancejador de càrrega. El balancejador de càrrega consulta el registre del servei i enllaça cada sol·licitud cap a una instància de servei disponible. Com en el cas del descobriment del client, les instàncies del servei es registren i es desregistren al registre del servei.

- ***Service registry*:** Aquest patró es basa en escollir l'estratègia de registre de serveis de manera que un servei puga descobrir un altre de manera efectiva. Aquest sistema es basa en una base de dades que conté les ubicacions de xarxa de les instàncies del servei. Cal que el registre de serveis estiga altament disponible i actualitzat. Alguns dels patrons més utilitzats són:

    - ***Self registration:*** Quan un servei es desplega al sistema es registra al sistema de manera que pot ser descobert pels altres serveis de la solució. Quan aquest servei es replega es desregistra del sistema, de manera que els altres serveis puguen saber que ja no està disponible. Si és necessari el propi servei ha d'utilitzar una instància pròpia per a generar un *hearth-beat* i evitar que el seu registre caduque.

    - ***3rd party registration*:** Quan utilitzeu un patró de registre de tercers, les instàncies del servei no són responsables de registrar-se al registre del servei. En canvi, un altre component del sistema conegut com a registrador de serveis gestiona el registre. El registre del servei fa un seguiment dels canvis al conjunt d'instàncies en execució mitjançant el mecanisme *hearth-beat* o la subscripció d'esdeveniments. Quan se n'adona que hi ha un nova instància de servei disponible, la registra al registre del servei. El registre del servei també cancel·la les instàncies del servei finalitzades.

2.2. Computació Autònoma

Durant els darrers anys, a causa de l'aparició de noves tecnologies (mòbils, computació al núvol, la IoT(*Internet of Things*), etc.) els sistemes han evolucionat a una velocitat vertiginosa. Hem arribat a un punt on cada vegada volem que els sistemes siguen més complets, eficients, fiables, versàtils, etc.

La Computació Autònoma és la capacitat d'un sistema per administrar-se a si mateix de forma automàtica, mitjançant tecnologies adaptatives que augmenten la capacitat d'adaptació i redueixen el temps que necessiten els professionals de les IT per a resoldre dificultats de sistema i efectuar altres tasques de manteniment [9, 10].

Les capacitats autònomes d'autogestió dels sistemes informàtics realitzen tasques que els professionals de les IT trien delegar en la tecnologia segons les polítiques preestablertes. La política adaptable determina els tipus de decisions i accions que realitzen les capacitats autònomes.

Les capacitats d'autogestió d'un sistema compleixen les seues funcions realitzant una acció adequada basada en una o més situacions que observen de l'entorn. La funció de qualsevol capacitat autònoma és un bucle de control que recull detalls del sistema i actua en conseqüència.

En un sistema autònom (auto-gestionable), l'humà té un nou rol: en comptes de controlar el sistema directament, aquest defineix les polítiques generals i les regles que guien el procés d'auto-adaptabilitat.

En un entorn autònom, els components del sistema, des dels dispositius fins al programari poden incloure la funcionalitat del bucle de control. Segons IBM per a que un sistema es considere autònom ha de complir quatre propietats o característiques [11]:

- ***Self-configuring*:** Es pot adaptar dinàmicament als entorns canviants:

Els components de tipus *Self-configuring* s'adapten dinàmicament als canvis en l'entorn, mitjançant polítiques proporcionades pel desenvolupador. Aquests canvis podrien incloure el desplegament de nous components o l'eliminació dels actuals, o canvis dramàtics en les característiques del sistema. L'adaptació dinàmica ajuda a garantir la resistència i la productivitat contínues de la infraestructura informàtica, donant lloc a un creixement i flexibilitat empresarial.

- ***Self-healing*:** Pot descobrir, diagnosticar i reaccionar a les alteracions:

Els components de tipus *Self-healing* poden detectar desperfectes del sistema i iniciar accions correctives basades en les polítiques preestablertes sense que això afecte a l'entorn informàtic. L'acció correctiva podria implicar una alteració del seu propi estat o produir canvis en altres components de l'entorn. El sistema informàtic en general es fa més resistent perquè les operacions quotidianes són menys propenses a fallar.

- ***Self-optimization*:** Permet controlar i ajustar els recursos automàticament:

Els components del tipus *Self-optimization* poden ajustar-se a les necessitats de l'usuari final o del negoci. Les accions d'ajust poden suposar l'optimització dels recursos disponibles o garantir que es poden realitzar certes accions en un moment determinat. L'auto-optimització ajuda a proporcionar un alt nivell de servei ja que assegura que els sistemes utilitzen els recursos de manera eficient i que certs recursos imprescindibles estaran disponibles en el moment de ser utilitzats.

- ***Self-protection*:** Pot anticipar, detectar, identificar i protegir-se contra amenaces des de qualsevol lloc:

Els Components de tipus *Self-protection* poden detectar comportaments hostils a l'hora que es produeixen i emprendre accions correctores per fer-se menys vulnerables. Les conductes hostils poden incloure accés i ús no autoritzats, infecció i proliferació de virus i atacs de denegació de serveis.

Com tota tecnologia, la computació autònoma te una sèrie de reptes associats que comentarem a continuació:

En primer lloc, la computació autònoma té l'objectiu de crear una solució de programari amb capacitats autònomes, és a dir, aconseguir traslladar la càrrega de la gestió de sistemes de les persones a les tecnologies. Aquest concepte implica que l'humà deixa de tindre control absolut sobre el sistema i passa a delegar responsabilitats sobre el sistema. Aquest problema es coneix com el problema del *control shift* [12] on els sistemes tenen cada vegada més responsabilitats, i a la vegada, el humans passen a tenir una responsabilitat de supervisió.

Açò suposa un repte ja que el programari ha de ser intel·ligible de manera que el responsable de les IT tinga control de la situació. Necessitem que el sistema autònom siga capaç de mostrar a l'usuari què està realitzant i quin és l'estat del sistema actual en temps real, de manera que l'usuari adquirisca confiança amb el sistema.

També hem de tindre en compte que la computació autònoma té un grau de complexitat elevat i per tant necessitem definir exactament quines són les característiques del sistema, els seus límits, etc [13, 14]. Definir l'entorn i el tipus de solució pot suposar un repte a l'hora d'utilitzar aquesta tecnologia i és probable que depenent de la solució que desitgem siga millor utilitzar altre tipus d'arquitectura o tècnica de desenvolupament de programari.

En primer lloc, és necessari definir amb exactitud l'espai de disseny i els límits d'aquest [20]. Han sorgit diverses metodologies parcials per identificar i representar els espais de disseny. En [15] defineixen un conjunt de dimensions de modelatge, i [16] es defineix un espai de disseny com a un conjunt de decisions que ha de prendre el programador. En [17] s'argumenten diferents propietats de disseny a l'hora de concebir els sistemes autoadaptatius.

D'aquestos treballs emana la idea d'identificar una sèrie de propietats de disseny relevants i representatives per al domini que es prtén abordar. Per tant que resulte més fàcil determinar el caràcter del sistema i la interacció entre les característiques d'autoadaptació i els bucles de control, seguirem amb el plantejament anterior on es comentava que l'humà delega les feines trivials al sistema. No obstant això, les feines amb un grau d'impacte alt haurien de ser gestionades pel tècnic d'IT per tant, el bucle de control i el sistema autoadaptatiu haurien de tindre un mecanisme de tipus jeràrquic on per realitzar una acció d'alt impacte sol·liciten confirmació a l'usuari.

Finalment, en un sistema autònom els entorns són dinàmics, és a dir, canvien els objectius del sistema. Com a conseqüència, necessitem mitjans adequats per comprendre plenament les característiques dels sistemes de programari autoadaptatius i les característiques clau dels seus cicles de vida per millorar el disseny, el modelat, l'optimització i la implementació d'aquests sistemes i processos.

A continuació s'explicarà dos conceptes rellevants de la computació autònoma, els sistemes autoadaptatius i els bucles de control MAPE-K.

2.2.1. Self-Adaptive Systems

Els sistemes autoadaptatius (SAS) són sistemes que poden desenvolupar-se i aprendre per si sols i en els quals tant els seus factors, com la seua estructura s'adapten a l'ambient en temps real, és a dir, poden canviar de forma autònoma el seu comportament com a resposta als canvis tant en els estats interns del sistema com en l'entorn. Aquests sistemes estan generalment associats amb sistemes que reben gran quantitat de dades i el mode d'operació és en temps real [9].

L'auto-adaptabilitat pot ser estàtica o dinàmica. En l'auto-adaptació estàtica, els mecanismes d'adaptació són definits explícitament pels dissenyadors perquè el sistema sàpiga quin d'aquests triar durant la seua execució, mentre que en l'auto-adaptació dinàmica, les adaptacions les produeixen, plans d'adaptació i requeriments observats durant la monitorització. Aquestes adaptacions són elegides pel sistema en temps real.

El caràcter propi de l'adaptació assegura un nivell mínim de supervisió humana, ja que l'humà necessita tindre un mínim de control sobre el sistema i saber que està fent mentre treballa de forma autònoma. A més, és necessari per a que les aplicacions que hagen de funcionar contínuament, fins i tot en condicions adverses, siguen capaces de canviar els requeriments sempre que siga necessari.

El paradigma dels SAS està basat en el terme d'evolucionar (expandir o bé reduir) l'estructura de sistema per tal que aquesta puga ser capaç d'adaptar-se als canvis en l'entorn. Açò pot suposar un repte ja que el sistema ha de ser capaç d'adaptar-se a les situacions dinàmicament i en temps real. Típicament, un SAS consisteix en un subsistema gestionat que implementa les funcionalitats del sistema i un subsistema gestor (autònom) que implementa la lògica d'adaptació.

Generalment, el subsistema del gestor s'implementa utilitzant el Model que va proposar IBM anomenat MAPE-K [11]. Aquest està constituït per un Monitor, un Analitzador, un Planificador i un Executor. Les quatre etapes estan suportades per una base de coneixement que permet l'intercanvi d'informació entre els components. D'aquesta manera, els SAS són capaços de desenvolupar l'estructura, la funcionalitat i la representació interna del coneixement. Aquesta evolució s'efectua des d'un aprenentatge progressiu, utilitzant les noves dades rebudes pels sensors del sistema i interactuant amb l'entorn per realitzar les adaptacions de la millor manera possible.

2.2.2. Bucle de control MAPE-K

El bucle de control MAPE-K va ser proposat per IBM i representa el model més popular. El bucle de control MAPE-K és el model de referència per dissenyar SAS en el context de la computació autònoma. La figura mostra un element autònom on el subsistema del gestor (autònom) implementa la lògica d'adaptació mitjançant el model MAPE-K i el sistema gestionat implementa les funcionalitats del sistema. Consta de sensors i efectors, així com quatre components que realitzen el bucle de retroalimentació: Monitor-Analitzador-Planificador-Executor, suportat per una base de coneixement que permet la transferència d'informació entre els altres components . Els sensors, adjunts al subsistema gestionat, controlen el subsistema gestionat per recollir dades que reflecteixen l'estat del sistema, mentre que els efectes apliquen els canvis al subsistema gestionat [18].

Aquestes quatre parts treballen conjuntament per proporcionar la funcionalitat del bucle de control. Les quatre parts es comuniquen i col·laboren entre elles i intercanvien coneixements i dades, tal com es mostra a la figura.

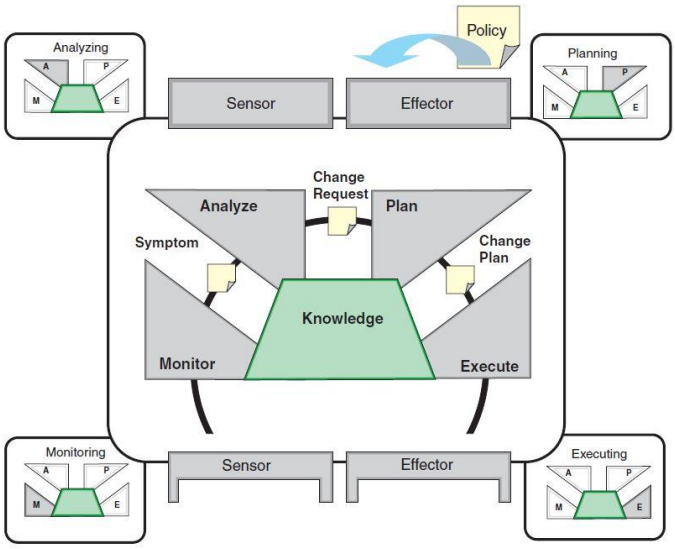

Figura 6: Bucle MAPE-K.

A continuació, passarem a detallar els elements que componen el bucle MAPE-K i la funció que exerceixen al subsistema gestor [11].

- **Funció Monitor:**

La funció de monitor recopila els detalls dels recursos gestionats a través dels sensors i els correlaciona amb símptomes que es poden analitzar. Els detalls poden incloure informació de topologia, mètriques, configuració del sistema, etc. Aquestes dades inclouen informació sobre la configuració dels recursos gestionats, l'estat, la capacitat i el rendiment. Algunes de les dades són estàtiques o canvien lentament, mentre que altres són dinàmiques, canviant contínuament al llarg del temps.

La funció del monitor afegeix, correlaciona i filtra les dades que rep dels sensors fins que detecta un símptoma que cal analitzar. Lògicament, aquest símptoma s'adreça a la funció d'anàlisi. El gestor autònom ha de recopilar i processar una gran quantitat de dades de la interfície del sensor d'un recurs gestionat.

- **Funció d'anàlisi:**

La funció d'anàlisi proporciona els mecanismes per observar i analitzar situacions per determinar si cal fer algun canvi; per exemple, si alguna política no s'està complint. La funció d'anàlisi és l'encarregada de determinar si el gestor autonòmic pot complir la política establerta, ara i en el futur. En molts casos, els models de funció analitzen un comportament complex, de manera que amb la informació aconseguida pot predir situacions i anticipar-se a situacions complicades que puguen acabar produint algun error al sistema.

Els gestors autonòmics han de ser capaços de realitzar anàlisis i raonaments de dades complexos sobre els símptomes que proporciona la funció de monitor. L'anàlisi està influenciada per dades de coneixement emmagatzemades.

Si es requereixen canvis, la funció d'anàlisi genera una sol·licitud de canvi i adreça aquesta sol·licitud de canvi a la funció de planificació. La sol·licitud de canvi descriu les modificacions que el component d'anàlisi considere necessàries o desitjables.

- **Funció planificador:**

La funció de planificació crea o selecciona un procediment per a realitzar una alteració desitjada en el recurs afectat per l'alteració del comportament detectada pel monitor. La funció de planificació pot adoptar moltes formes, que van des d'un sol canvi al sistema a un flux de treball complex.

La funció de planificació genera el pla de canvis adequat, que representa un conjunt desitjat de canvis per al recurs afectat, i el qual s'adreça a la funció d'execució per a que realitze els canvis pertinents.

- **Funció d'execució:**

La funció d'execució proporciona el mecanisme per planificar i realitzar els canvis necessaris al sistema. Una vegada que un gestor autònom ha generat una sol·licitud de canvi i l'executor ha rebut el pla d'acció la funció d'execució és l'encarregada de dur a terme el procediment per aplicar el pla d'acció al servei afectat. Aquestes accions es realitzen mitjançant la interfície d'efector d'un recurs gestionat. Una part de l'execució del pla d'acció podria implicar l'actualització del coneixement que s'utilitza pel gestor autònom.

- **Funció de coneixement:**

Una font de coneixement és la implementació d'un registre, diccionari, base de dades o un altre repositori que proporciona accés al coneixement d'acord amb les interfícies prescrites per l'arquitectura. En un sistema autònom, el coneixement consisteix en un tipus particular de dades amb sintaxi i semàntica guardada, com ara símptomes, polítiques, peticions de canvis i plans de canvis. Aquest coneixement es pot emmagatzemar en una font de coneixement perquè es puga compartir entre gestors autònoms.

El coneixement emmagatzemat en fonts de coneixement es pot utilitzar per ampliar les capacitats de coneixement d'un gestor autònom. Un gestor autònom pot carregar coneixement d'una o més fonts de coneixement per fer comparacions i decidir quan o com aplicar decisions sobre el sistema.

2.3. Conclusions de l'Estat de l'Art

Aquestes tecnologies d'avantguarda ens permeten poder gestionar les solucions de programari de manera més còmoda ja que en estar muntades amb diversos components, podem actualitzar, substituir, afegir/retirar, etc. els components que necessitem en un precís moment i així aconseguir una vida útil major que amb els mètodes tradicionals.

A més, aquestes tecnologies permeten reduir la càrrega de treball del professional d'IT ja que mitjançant computació autònoma podem delegar feines trivials al sistema. Aquesta tecnologia està utilitzant-se cada vegada més però podríem aconseguir una solució un poc millor que les que hi ha actualment.

Si observem el panorama actual, hi ha moltes solucions que empren o bé computació autònoma o bé µserveis, però en rares ocasions podem veure sistemes que empren aquestes dues tècniques a la vegada. Realment és tan complicat fusionar les dos tècniques o es podrien utilitzar ambdues a la vegada i així augmentar el nivell d'escalabilitat d'un sistema?

Aquest treball pretén assolir un enfocament on donem un pas més endavant i integrem ambdues tècniques de manera que aconseguim un sistema que, no sols és capaç d'escalar afegint més µserveis, sinó que és capaç d'afegir, eliminar, modificar, etc. components de la solució per optimitzar els recursos en temps real i, a més, autogestionar-se de manera que aplique diferents estratègies per resoldre les incidències que ens podríem trobar durant l'execució de la solució, com ara errors de comunicacions, sobrecàrrega del sistema, augment en les peticions a un µservei, etc.

A més, si veiem l'estat actual de l'Arquitectura Orientada a µServeis (SOA) ens trobem que no hi ha una notació estàndard. Diversos autors han fet propostes sobre aquest tema i han definit diverses notacions però, no hi ha cap que es puga considerar com la notació estàndard per als µserveis. Nosaltres emprarem la notació proposada pel tutor d'aquest treball que ve derivada de la que va proposar Martin Fowler.

3. Enginyeria de µserveis autònoms

Per a desenvolupar les capacitats de computació autònoma emprarem l'aproximació FADA (desenvolupada al centre de recerca PROS de la UPV, al grup TaTami) que permet realitzar l'enginyeria de sistemes auto-adaptatius. És un enfocament de desenvolupament dirigit per models, on a través de manipulacions sobre els mateixos (emprant tècniques que provenen del desenvolupament dirigit per models o *Model Driven*), permeten establir els requeriments i desenvolupar solucions funcionals. Aquesta aproximació empra models tant per a la fase d'enginyeria, com per a la fase d'execució (models en temps d'execució o m@rt). En temps d'execució els models són emprats pel propi sistema per a prendre decisions i mantenir-se sempre en un estat coherent. Així doncs, el primer pas que necessitem per a desenvolupar µserveis autònoms és emprar alguna notació abstracta que ens permeta capturar la natura de l'aplicació. Per a fer-ho, emprarem en aquesta tesina un metamodel que han desenvolupat en aquest centre de recerca.

Després, s'hauran de definir un conjunt d'operacions de manipulació sobre aquest metamodel amb l'objectiu de definir les capacitats de canvi que volem establir sobre aquest tipus de solucions basades en µserveis. En aquest treball, explorarem l'aplicació d'un subconjunt dels patrons explicats anteriorment que s'utilitzen al món del desenvolupament de solucions basades en µserveis i que defineixen un conjunt de configuracions típiques per abordar l'enginyeria d'aquest tipus de solucions.

Finalment, descriurem com hem integrat tot açò dins d'un bucle d'adaptació MAPE-K. Aquest bucle MAPE-K, amb el conjunt de recursos que hi definirem, ens establirà el marc de treball que permetrà dotar a la solució de µserveis de capacitats d'auto-gestió. A través de sondes, monitors i regles d'adaptació (entre altres recursos), dissenyarem una solució que puga monitoritzar, decidir i canviar la solució de µserveis sobre la que s'aplique.

3.1. Llenguatge de descripció de solucions basades en µserveis

Un "metamodel" és un model abstracte que representa els conceptes d'un domini concret. Els metamodels són artefactes essencials en les estratègies de desenvolupament basades en models, ja que la instància d'un metamodel és un model que representa alguna solució concreta. Aquest metamodel es converteix, doncs, en un llenguatge abstracte per a descriure eixos conceptes del domini. Per exemple, al metamodel d'UML es descriuen els diferents conceptes (classes, relacions, actors, casos d'ús, etc.) que constitueixen la notació que UML proposa per a fer anàlisis de sistemes Orientats a Objectes.

Aquests llenguatges solen vindre acompanyats de notacions concretes o representacions, que descriuen maneres d'emprar aquests llenguatges per tal d'obtenir les especificacions dels sistemes. Aquestes notacions poden ser textuals (com podria ser un llenguatge de programació), o gràfiques (com podria ser una notació típica d'un Diagrama de Classes d'UML).

De la mateixa manera, necessitem emprar un metamodel de referència que ens descriga els conceptes d'una solució basada en µserveis. Al Centre PROS s'ha treballat en un metamodel (és producte d'una recerca en curs), i que es mostra a la següent figura:

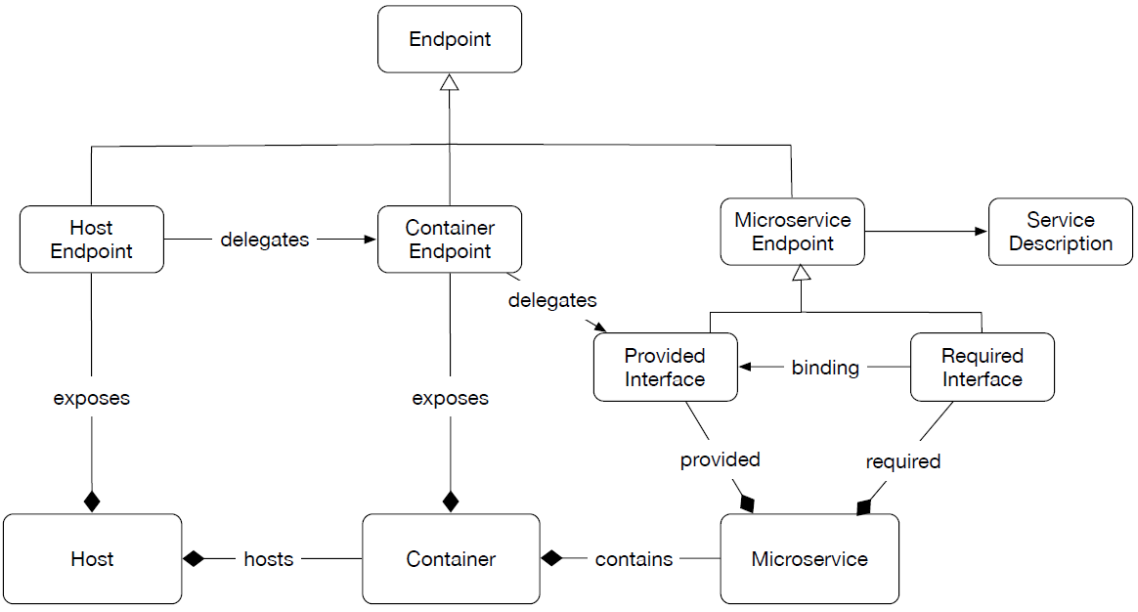

Figura 7: Metamodel de referència per a descriure solucions amb µserveis.

Com es pot observar, al metamodel apareixen relacionats els diferents conceptes d'una solució basada en µserveis. A continuació es descriu aquest metamodel i la seua notació gràfica associada.

3.1.1. µserveis

Un **µservei (servei o aplicació)** és l'abstracció d'una peça de programari que implementa una funcionalitat i que es pot interconnectar amb altres µserveis per oferir una determinada funcionalitat més complexa. Gràficament es representen com cercles, i són el cor d'una solució basada en µserveis. Quan un µservei ofereix una funcionalitat i es comunica amb altres µserveis que ofereixen altres funcionalitats, direm que són part d'una solució de programari.

Un µservei exposa una sèrie d'interfícies com a mecanismes de comunicació amb altres. Pot definir dos tipus d'interfícies:

- **Interfície proporcionada (⚲):** Representa un servei que el µservei proporciona, i defineix un "tipus" d'interfície o contracte, que aquest µservei proporciona.

- **Interfície requerida (⅄):** Representa un servei que el µservei necessitarà per a funcionar i que un altre µservei li haurà de proporcionar. Indica el "tipus" d'interficie o contracte que requereix.

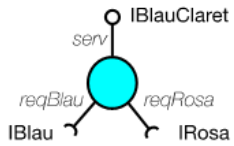

Figura 8: µservei amb interfícies.

Cada interfície que un µservei proporciona o requereix va, a més a més, etiquetada amb un nom d'interfície, per a poder identificar-la i distingir-la de la resta.

A la figura 8 es pot veure un µservei que defineix una interfície proporcionada (de nom "serv" i tipus d'interfície "IBlauClaret"), i dues interfícies requerides, una anomenada "reqRosa" de tipus "IRosa" i l'altra "reqBlau" de tipus "IBlau". Amb aquesta especificació, en està dient que per a que aquest µservei puga funcionar, requerirà que existisquen altres µserveis que proporcionen les interfícies "IRosa" i "IBlau". Ell, per la seua part, ofereix la interfície "IBlauClaret" per a que altres µserveis puguen emprar-la, si la requereixen.

3.1.2. Enllaços

En una solució orientada a µserveis, aquests estableixen relacions entre ells a través de les seues interfícies proporcionades i requerides. Els **enllaços, vincles o *bindings* (⟶)** representen una connexió entre una interfície que un µservei requereix amb una interfície "compatible" (del mateix tipus) que un altre µservei proporciona. Aquest enllaç defineix una dependència direccional entre µserveis i s'empra per a configurar la solució.

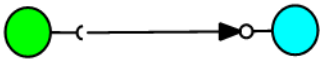

Figura 9: Enllaç entre µserveis.

En la figura 9 es pot veure un enllaç entre dos µserveis. La interfície requerida pel µservei verd està enllaçada amb la interfície proporcionada pel µservei blau clar.

3.1.3. Especificació lògica d'una aplicació de µserveis

Amb aquesta notació ja és possible descriure solucions basades en µserveis de manera que es defineixen les seues interfícies d'operació (i dependències) i s'estableixen els vincles (*bindings*) existents entre ells. A la figura 10 es pot veure una solució formada per cinc µserveis on per exemple, el µservei verd requereix una interfície mentre que el µservei blau n'exposa una.

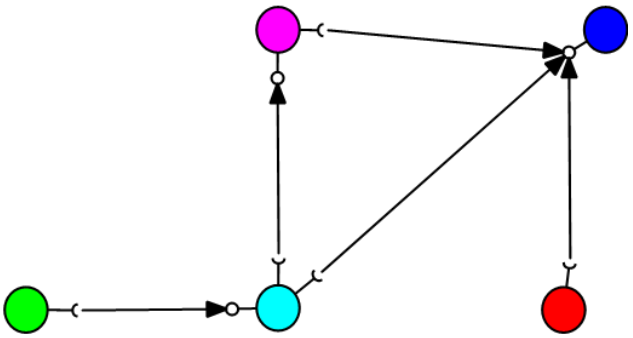

Figura 10: Esquema lògic d'una aplicació de µserveis.

Amb aquesta notació, som capaços de representar una solució abstracta o lògica basada en µserveis, però necessitem ser capaços de representar més informació relacionada amb la manera en la que l'aplicació es desplega sobre una infraestructura computacional completa. En aquest punt entren en joc els conceptes de contenidors i nodes computacionals (o *hosts*), que són la plataforma computacional "natural" d'una solució basada en µserveis.

3.1.4. Node computacional

Un **Node computacional o *host*** és una dispositiu capaç d'executar serveis i aplicacions, sent aquest dispositiu físic o digital. Aquest *host* contindrà una sèrie de contenidors que utilitzarem per a encapsular els diversos µserveis de la nostra solució. En execució, un host podrà ser una màquina física o una màquina virtual que allotge una o vàries solucions mitjançant contenidors. Gràficament el representarem com:

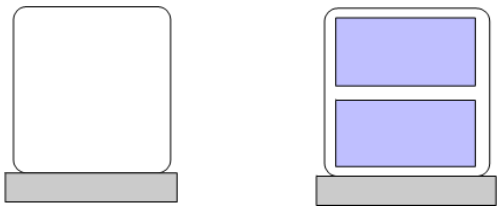

Figura 11: Node computacional (esquerre) i Node computacional amb dos contenidors (dreta).

3.1.5. Contenidor

Un **contenidor** és un servei especial capaç de contenir i executar µserveis/aplicacions. Encapsula aplicacions, ocultant-los de la resta de nodes de computació, contenidors i/o µserveis. Es poden establir mecanismes, a través de **ports de comunicació i delegacions**, per habilitar l'ús dels recursos disponibles dintre del contenidor. Gràficament el representarem com:

Figura 12: Contenidor (esquerre) i Contenidor amb un µservei (dreta).

3.1.6. *Endpoints* i delegacions

Per a que els µserveis puguen comunicar-se necessitem *endpoints* i delegacions ja que els contenidors i els recursos aïllen els µserveis que contenen de la resta. A continuació explicarem en què consisteixen aquests dos recursos i com es modelen. Un ***enpoint* o punt de connexió (□)** és un mecanisme que permet connectar µserveis entre *hosts* o contenidors i una **Delegació (----➔)**, és un mecanisme semblant als enllaços que ens permet utilitzar els diferents canals de comunicació que s'habilitaran per interconnectar recursos (contenidors i nodes de computació).

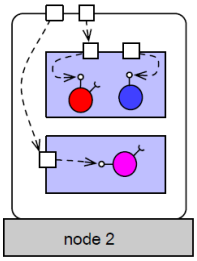

Figura 13: *Endpoint* i delegacions.

3.1.7. Tipus d'enllaços

Per acabar, reprenem el concepte d'enllaç (⟶) i en definim de tres tipus que existeixen en funció de com s'establisca l'enllaç:

- **Enllaç privats:** Ocorren entre dos µserveis allotjats dins del mateix contenidor. Un µservei es connecta a l'altre sense necessitat de connectar-se a *endpoints* exposats per contenidors ni hosts. En la següent figura és el cas de l'enllaç entre el µservei roig i blau, dins el contenidor de superior en el node 2.

- **Enllaç locals:** Ocorren entre dos µserveis allotjats en contenidors diferents dins del mateix node computacional. Requereix que el µservei que ofereix la interfície la tinga exposada en el seu contenidor a través d'un *endpoint*, i que aquest *endpoint* estiga delegat per poder accedir a la interfície amb la que necessitem comunicar-se. El µservei que requereix la interfície s'ha d'enllaçar a ell a través de *l'endpoint* exposat pel contenidor. En la següent figura és el cas dels enllaços entre el µservei rosa i el blau en el node 2 o entre el µservei verd i el blau dins el node 1.

- **Enllaç públics**: Ocorren entre µserveis allotjats en nodes computacionals diferents. El µservei que ofereix la interfície ha d'exposar-la (i rebre una delegació) en el contenidor on es troba, i a la seua vegada, el node computacional ha d'exposar també un *endpoint* i delegar-lo sobre *l'endpoint* definit al contenidor. D'aquesta manera, s'habilita un circuit de delegacions de manera que, a través de *l'endpoint* del node, es redirigeix finalment a la interfície exposada pel µservei. El µservei que requereix la interfície estableix un enllaç amb *l'endpoint* exposat al node computacional. En la figura és el cas de l'enllaç entre el µservei verd i *l'endpoint* del node 2 que acaba delegant-se sobre la interfície del µservei rosa.

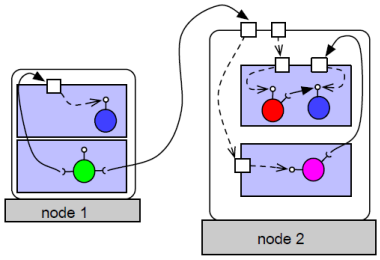

Figura 14: Tipus d'enllaços.

3.1.8. Especificació del desplegament d'una aplicació de µserveis

Segons s'ha comentat abans, la figura 10 mostra una vista lògica d'una aplicació de µserveis, on es veu com està configurada (quins µserveis i quins enllaços hi ha establerts), però de manera abstracta respecte a la infraestructura computacional. Per tal de mostrar com es podria realitzar el desplegament sobre una infraestructura disponible ja concreta, s'empra una altra especificació on apareixeran definits explícitament: contenidors, hosts, *endpoints* i delegacions.

Aquesta especificació de desplegament, descriu una possible configuració operacional per a una vista lògica, emprant uns recursos concrets. Per a una mateixa vista lògica, poden existir molts desplegaments diferents, capturant d'aquesta manera, aquesta possibilitat real.

Si prenem com a exemple la solució descrita a la figura 10, podem pensar que la podríem desplegar de la següent manera:

- Allotgem cada µservei a dintre d'un contenidor diferent.
- Per cada interfície proporcionada que oferisca un µservei, es crea un *endpoint* a nivell del contenidor i una delegació cap a aquesta interfície proporcionada.

- Es creen nodes computacionals.
- S'allotgen/reparteixen els contenidors dins els nodes computacionals. Quan un contenidor s'ubica en un node computacional, per a cada *endpoint* que ofereix el contenidor, es crea un *endpoint* al node computacional i una delegació cap a *l'endpoint* del contenidor.
- Per cada enllaç definit a la vista lògica es crea un enllaç privat, local o públic en funció de com es troben allotjats ambdós µserveis de l'enllaç.

Seguint aquesta heurística, i assumint que tenim disponibles 4 nodes computacionals (node 1, node 2, node 3 i node API-gateway), podem arribar a tenir un desplegament com el que es mostra a la figura 15, on:

- Cada µservei s'ubica en un contenidor (anomenats del contenidor c1 fins al contenidor c5)
- Als contenidors c2, c3, c4 i c5 s'han creat *endpoints* (8080, 9001, 9002 i 9001, respectivament) amb les seues delegacions a les interfícies proporcionades pels seus µserveis.
- Els contenidors s'han repartit entre els nodes computacionals, i per a cada *endpoint* que proporciona un contenidor, s'ha creat el seu homòleg al host, sobre el que s'ha realitzat també la delegació.
- Cada enllaç s'ha establert emprant els *endpoints* disponibles. Com que ha coincidit que tots els µserveis requereixen interfícies d'altres µserveis allotjats en nodes diferents, en aquest cas, tots els enllaços s'han materialitzat com enllaços públics sobre *l'endpoint* que exposa el node computacional que acaba delegat sobre la interfície del µservei proporcionat adient (a través de *l'endpoint* i delegació al contenidor).

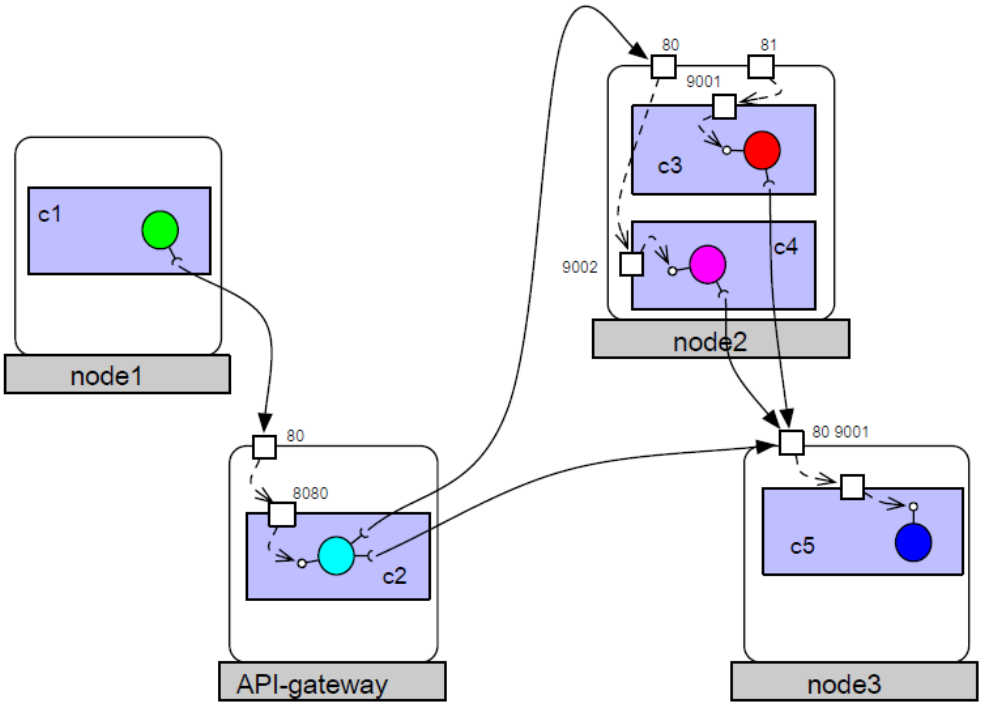

Figura 15: Esquema del desplegament.

3.2. Patrons aplicats als µserveis

El primer pas per a fer el disseny d'aquest tipus de solucions és imaginar que nosaltres fem la funció del bucle de control MAPE-K i, per tant, les decisions les prenem nosaltres. Si nosaltres som el bucle i tinguérem el sistema inicial descrit anteriorment, obtindríem una solució com la que s'espera obtindre però manual, ja que nosaltres fem el paper del bucle.

Si hi haguera una incidència amb un component el que faríem seria desconnectar el component que ens està donant problemes i substituir-lo per un altre. Si ara, per exemple, observàrem que un *host* està sobrecarregat i necessitàrem disminuir la càrrega, el que faríem seria desconnectar alguns components, dur-los a una altra màquina i reconnectar-los.

Doncs, seguint aquesta mecànica on el que fem per a aconseguir les adaptacions és desconnectar, connectar, crear, eliminar, moure, etc. el que farem serà, emprant l'enfocament de desenvolupament FADA del Centre PRoS que ens proporciona mecanismes per a implementar aquestes accions, crear classes noves, crear les proves, els monitors, les sondes, etc. i dissenyar una proposta de solució on el bucle MAPE-K aplique aquests patrons.

Per fer l'anàlisi dels diferents patrons, explicar com funcionen i veure com s'aplicarien a qualsevol escenari/especificació feta amb la notació proposta, utilitzarem l'escenari del capítol anterior (veure Figura 15). A banda, del conjunt de patrons de µserveis disponibles, hem fet una selecció centrant-nos en aquells que solen emprar-se habitualment, i que la seua implicació siga allò més clara possible. En aquest treball es pretén donar un primer pas cap endavant per ajudar a dissenyar solucions de µserveis amb capacitats de computació autònoma, pel que, el nostre objectiu no és cobrir tot l'espectre de patrons, sinò obrir el camí per a poder aplicar la resta, o inclús altres. Tot açò seguint una estratègia de disseny semblant a les proposades en [19, 20], els patrons seleccionats han sigut:

- ***Load Balancer*:** Permet desplegar o replegar el patró *Load Balancer* sobre un µservei depenent de la necessitat del sistema.
- ***Circuit Breaker*:** Permet desplegar o replegar el patró *Circuit Breaker* sobre un µservei depenent de la necessitat del sistema.
- ***Service per Container*:** Aplica el patró *Service per Container* a la solució.

A més a més, seguint una estratègia similar al treball realitzat en [21], s'han tingut en compte una sèrie d'operacions de manipulació més elementals, que hem identificat que són la base per a construir aquests i altres possibles patrons. Aquestes funcions, ens permetran modificar, crear i eliminar els diferents components del sistema de manera que ens ajudaran a realitzar les possibles adaptacions que requerisca el sistema per funcionar adequadament.

En les següents seccions veurem com funcionen tant els patrons com aquestes altres operacions, quines característiques posseeixen i com s'aplicarien a escenaris concrets.

3.2.1. Load Balancer

Per fer el patró *Load Balancer*, crearem un µservei que realitze la funció del *Load Balancer* (µservei lila) i l'encapsularem a un contenidor (contenidor C6), per desplegar-lo posteriorment a un *host* (*host* "Load Balancer"). Aquest µservei exposarà una interfície i en requerirà una altra. A més, hem d'afegir els *endpoints* (*enpoints* 8100) i delegacions necessàries perquè es puga comunicar amb els altres µserveis de la solució. Una vegada tenim el µservei creat i configurat, el desplegarem al sistema (ressaltat en roig).

Aquest µservei serà el que es col·locarà d'intermediari i mitjançant un algoritme de balanceig decidirà a quina de les instàncies anirà dirigida la petició. D'aquesta manera aconseguim reduir considerablement la càrrega de la instància a la qual li hem aplicat el *Load Balancer*.

Una vegada tenim preparat i desplegat el *Load balancer*, necessitarem desconnectar, copiar i desplegar rèpliques del µservei que estava sobrecarregat (µservei blau). Desplegarem una quantitat de còpies o una altra depenent de la sobrecàrrega que tinguera el sistema abans d'aplicar el patró. En aquest cas, hem fet dues còpies del µservei i les hem desplegat a *hosts* diferents, col·locant una còpia per *host* (nodes 4 i 5) a banda de la que ja teníem (node 3), per tindre un total de tres instàncies del mateix µservei (ressaltat en blau).

En aquest cas, com generem *hosts* nous necessitem que siguen idèntics a l'original o continguen els mecanismes necessaris per comunicar-se, és a dir, a banda del µservei hem de copiar els *enpoint* i les delegacions per a garantir que es poden comunicar.

Finalment, desplegarem les còpies i tornarem a fer els enllaços per a que el sistema puga comunicar-se amb el µservei blau i viceversa. Necessitarem aplicar sis enllaços, un del µservei blau clar al *Load Balancer*, un del µservei roig al *Load Balancer*, un del µservei morat al *Load Balancer*, un del *Load Balancer* al µservei blau del node 3, un del *Load Balancer* al µservei blau del node 4 i un del *Load Balancer* al µservei blau del node 5.

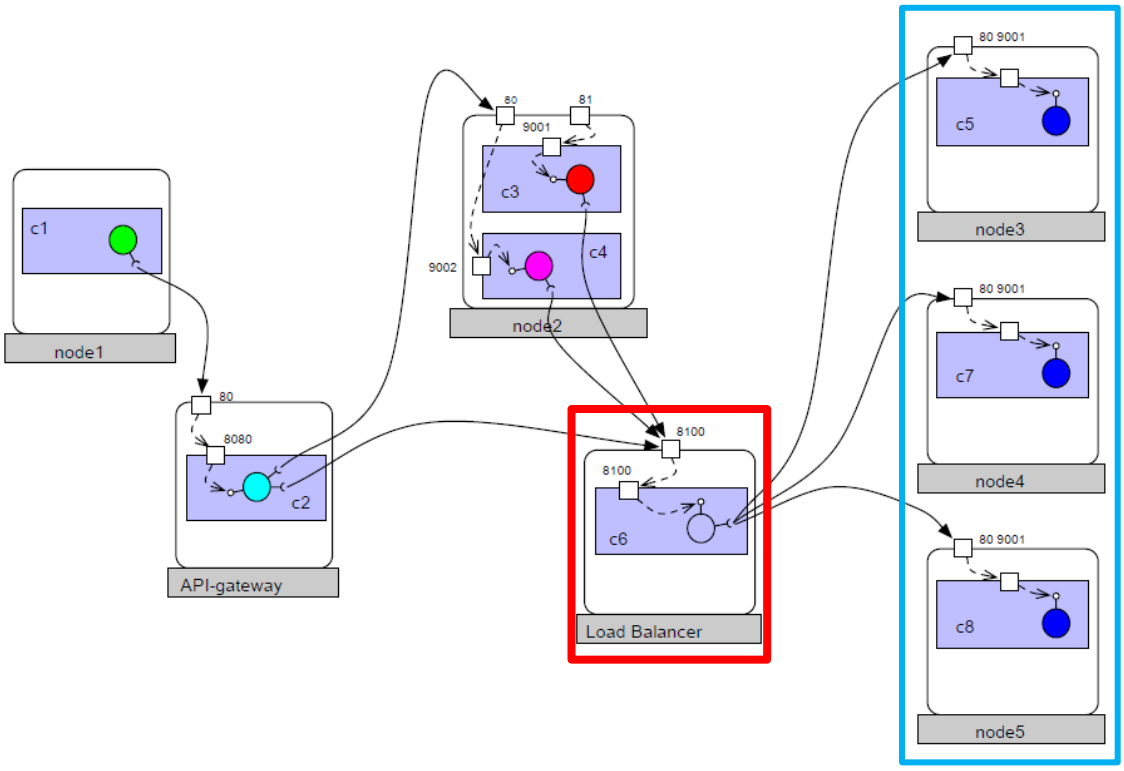

Figura 16: Patró *Load Balancer*.

3.2.2. Circuit Breaker

Per fer el patró *Circuit Breaker*, crearem un µservei que realitze la funció del *Circuit Breaker* (µservei lila) i l'encapsularem en un contenidor per desplegar-lo posteriorment a un *host* (*host* "Circuit-breaker"). Aquest µservei exposarà una interfície i requerirà una altra. A més, hem d'afegir els *endpoint* (*endpoints* 8100) i delegacions necessàries per que es puga comunicar. Una vegada tenim creat i configurat el *Circuit Breaker*¸ el desplegarem al sistema (ressaltat en roig).

Aquest µservei serà el que es col·locarà d'intermediari i mitjançant un algoritme de control d'incidències decidirà si un µservei està funcionant correctament o ha d'entrar en acció. Per saber quan ha d'entrar en acció, s'estableixen uns llindars, de manera que, si són superats, el *Circuit Breaker* s'activarà i aïllarà el node en qüestió (ressaltat en blau).

Una vegada tenim preparat el patró, desconnectem el µservei a aplicar-li el patró (µservei blau) i reconstruïm les connexions, en aquest cas, necessitarem 4 enllaços: un del µservei blau clar al *Circuit Breaker*, un del µservei roig al *Circuit Breaker*, un del µservei morat al *Circuit Breaker* i un del *Circuit Breaker* al µservei blau del node 3.

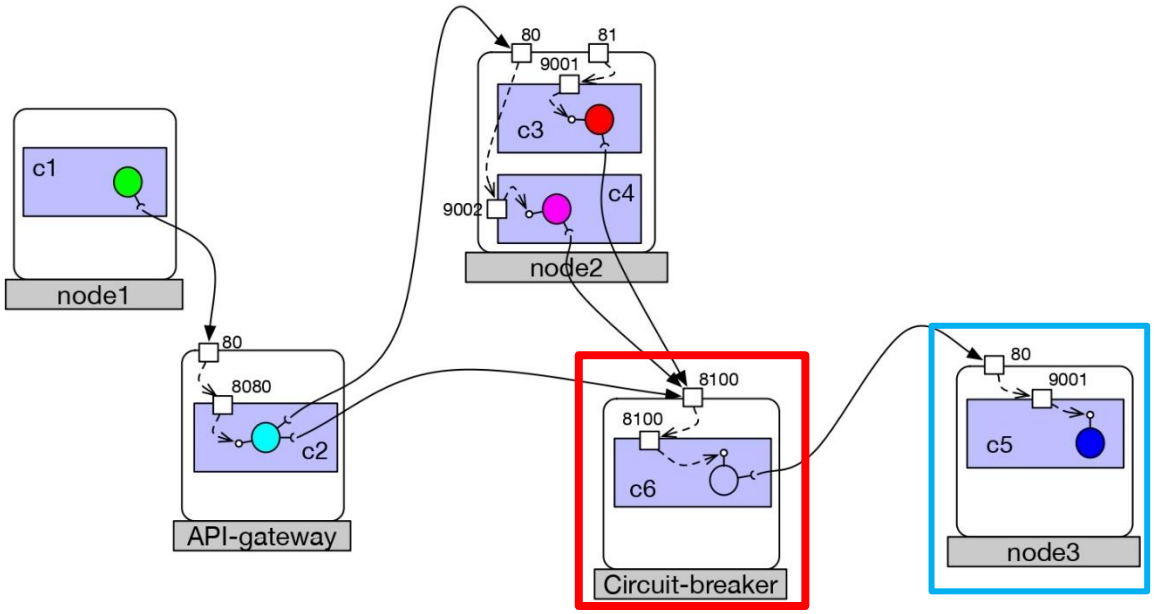

Figura 17: Patró *Circuit Breaker*.

3.2.3. Service per Container

Per fer el patró *Service per Container* farem un anàlisi de tot el sistema per veure si trobem algun contenidor que conté més d'un µservei. Si aquest és el cas haurem de moure'l. Aquest patró el que busca és deixar un sistema amb un cert patró de desplegament de manera que no ens trobem que hi ha una barreja de patrons i no segueixen cap lògica.

Per veure si es compleix o no, el que farem serà analitzar tots els contenidors del sistema (contenidors C1, C2, C3, C5), una vegada analitzats si hem detectat algun contenidor amb més d'un µservei (contenidor C3), haurem de moure a un contenidor nou tots els µserveis necessaris fins que sols quede un per contenidor.

En aquest cas com sols tenim un contenidor que conté més d'un µservei (ressaltat en roig), sols haurem de moure els µserveis que conté per a que es complisca el patró de desplegament. Si, per exemple, moguem el µservei morat, el que faríem seria desconnectar-lo i encapsular-lo a un contenidor nou (contenidor C4 figura 15) i moure el *endpoint* corresponent a aquest µservei (*endpoint* 9002) al nou contenidor.

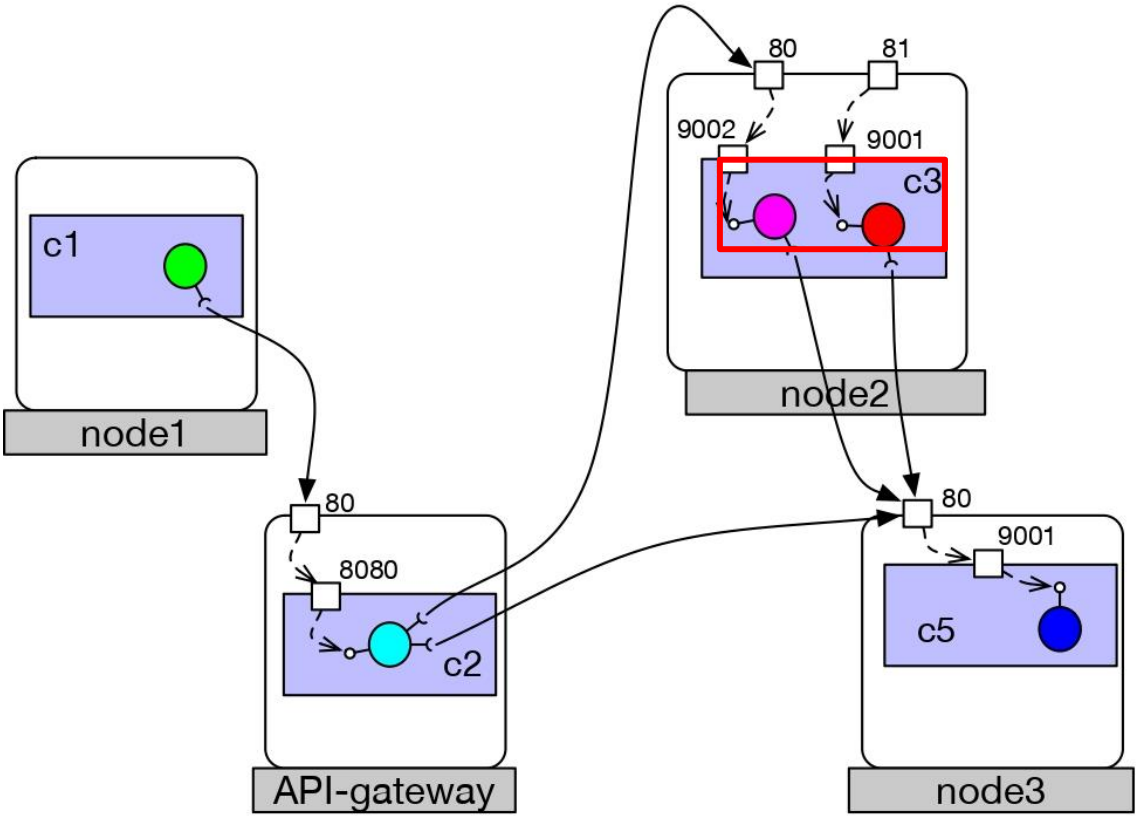

Figura 18: Patró *Service per Container*.

3.3. Altres operacions

A més dels patrons explicats anteriorment, també comptem amb una sèrie de funcionalitats que ens ajudaran a l'hora de realitzar adaptacions del model. Aquestes funcions ens serviran principalment per modificar els recursos de la solució de manera que, si necessitem afegir eliminar o modificar algun recurs, ho farem utilitzant aquestes funcions.

- **Moure contenidors/ µserveis:** Ens permet moure contenidors i µserveis d'un *host* a un altre.
- **Copiar Contenidors/ µserveis:** Ens permet copiar un µservei o un contenidor.
- **Esborrar *Hosts*/ Contenidors/ µserveis:** Ens permet esborrar recursos.

Aquestes funcions tenen una peculiaritat a tindre en compte que comentarem més avant, és tracta de la modificació en cascada. Aquesta peculiaritat es basa en què no podem eliminar o moure un element si dins d'aquest hi ha diversos elements. Per poder moure un element que conté elements, necessitem moure l'element més intern per deixar buit el contenidor i així poder moure'l.

3.3.1. Moure recursos

Per moure un recurs (en aquest cas un µservei) necessitarem desconnectar-lo i fer l'operació de moviment. En aquest cas mourem el µservei morat del node 2 al node 3. Per moure el µserveis farem una còpia temporal d'aquest. Una vegada tenim la còpia, generarem un contenidor nou (contenidor C4) al node 3 i eliminarem el contenidor C4 del node 2.

A continuació, desplegarem la còpia al contenidor C4 del node 3 i tornarem a generar les delegacions i els *endpoints* (*endpoints* 81 i 9002) per a que es puga comunicar. Finalment tornarem a fer les connexions necessàries, en aquest cas sols necessitarem crear un enllaç, que anirà del µservei blau clar del node "API-Gateway" al node 3.

Cal destacar que si, per exemple, necessitarem moure un contenidor sencer amb diversos µserveis o un *host* sencer amb diversos contenidors i µserveis, hauríem de fer-ho aplicant el moviment en cascada, és a dir, hauríem de moure primer el µserveis que es troben dins dels contenidors i finalment els contenidors que es troben dins del *host*.

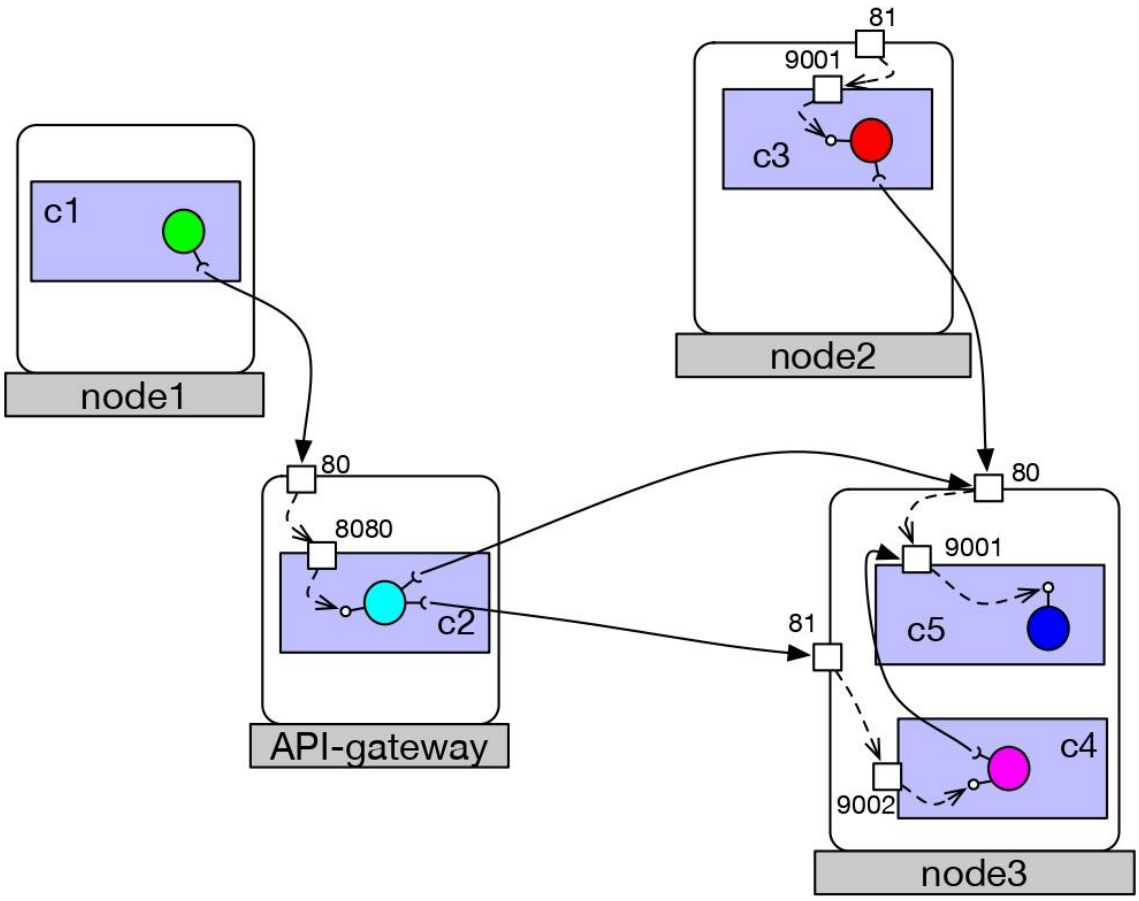

Figura 19: Funció moure recursos.

3.3.2. Copiar recursos

Per copiar els recursos, aplicarem una tècnica similar a la de moure que hem explicat abans. El que hem de fer és desconnectar el recurs que volem copiar, en aquest cas el µservei blau. Una vegada el tenim desconnectat sols necessitarem crear un contenidor nou (contenidor C6) en la ubicació que desitgem, en aquest cas el node 3.

Finalment desplegarem el µservei al contenidor nou i crearem els *enpoints* (*enpoints 81 9002*) i delegacions necessàries per a que es puga comunicar. A més, haurem de crear les connexions necessàries, en aquest cas, necessitem tres enllaços, un del µservei blau clar al µservei blau, un del µservei roig al µservei blau i un del µservei morat al µservei blau.

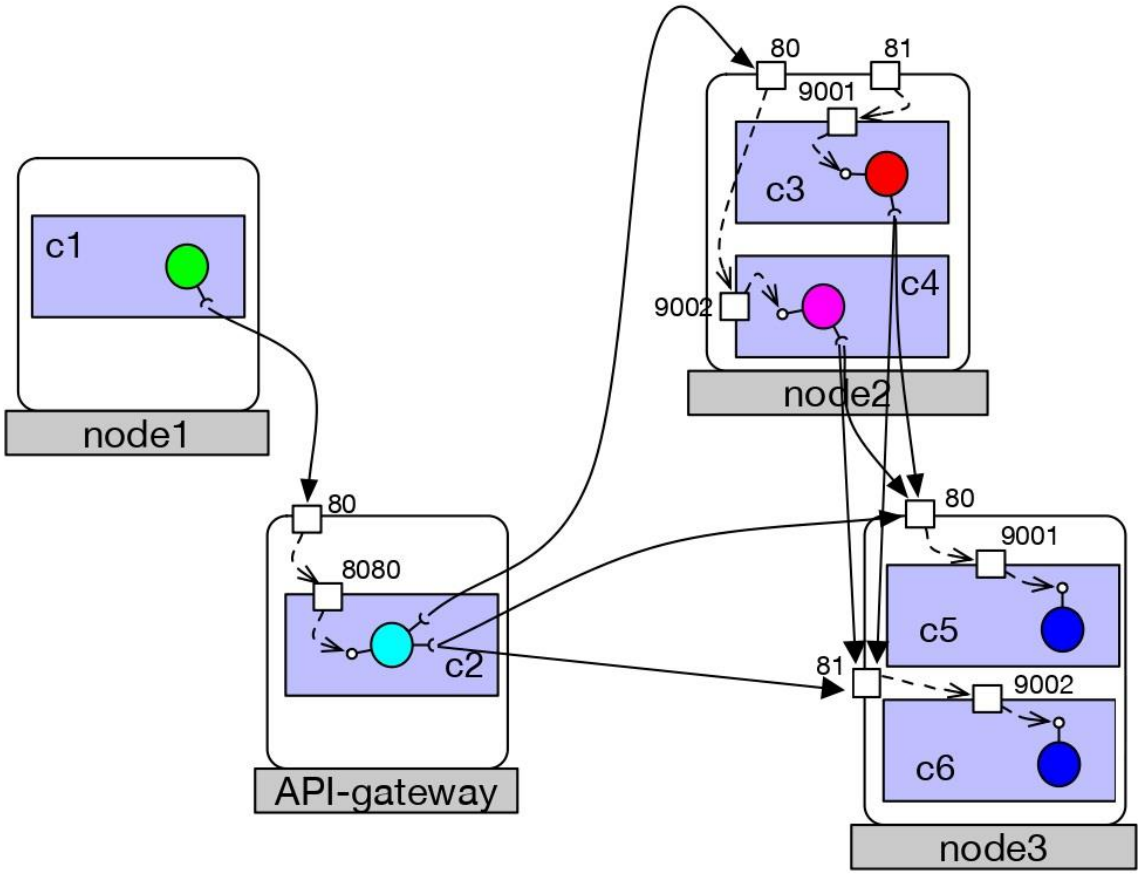

Figura 20: Funció copiar recursos.

3.3.3. Eliminar recursos

Per esborrar tots els recursos d'un *host*, necessitarem aplicar un esborrament en cascada, ja que hem d'esborrar primer els contenidors i per esborrar els contenidors hem d'esborrar abans els µserveis, de la mateixa manera passava en la funció de moviment o de copiat de recursos.

En aquest cas, hem esborrat el contenidor C3, per tant el que farem serà esborrar primer els µserveis que continga aquest contenidor (µservei roig) i després esborrar el contenidor en qüestió. Per eliminar el µservei roig, l'únic que hem de fer és desconnectar-lo de la solució i indicar-li al sistema que per a la següent adaptació que realitze l'ha d'esborrar. Una vegada hem dit que ha d'esborrar el µservei, li direm al sistema que esborre tant el contenidor C3 com els possibles *endpoint* o delegacions que continga.

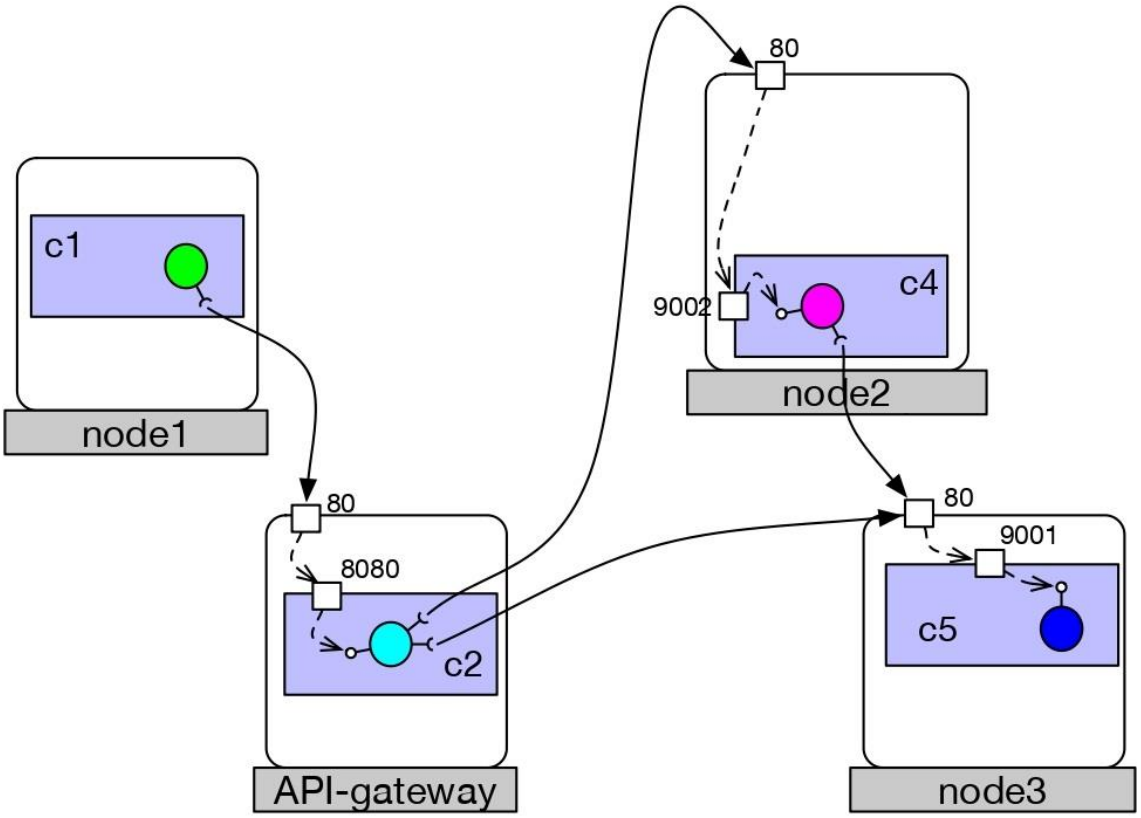

Figura 21: Funció eliminar recursos.

3.4. Enginyeria del bucle de control

Per a que els patrons descrits anteriorment funcionen correctament i el sistema siga capaç d'autoregular-se, hem de pensar en una sèrie d'elements addicionals que siguen responsables d'identificar i detectar quan és necessari aplicar alguna correcció a través de l'aplicació dels patrons i operacions abans detectades.

Per a aconseguir-ho, necessitarem definir, per a cada patró o acció correctiva el següent:

- Tipus d'informació/mesures que han de controlar-se i que són rellevants per a aplicar el patró.
- Els condicionants que s'han d'esdevindre per a poder aplicar-lo.
- Els patrons que s'haurien d'aplicar i què hauria de fer.

Per a descriure-ho, hem emprat les següents plantilles, basades en treballs fets en l'assignatura 'Disseny de Sistemes Ubics i Auto-Adaptatius' del màster MITSS del DSIC.

3.4.1. Requeriment d'adaptació: µserveis amb càrrega de treball variable

En aquest cas, passarem a descriure el requeriment d'adaptació per al balancejador de càrrega. Aquest s'aplicarà seguint els criteris descrits a continuació, sempre i quan es complisquen certes condicions. La finalitat d'aquesta acció correctiva és disminuir la càrrega computacional que està sofrint un µservei en un determinat moment.

Load Balancer	
Requeriment adaptació	Aplicar estratègia de balanceig.
Inputs d'informació o símptomes	Nivell de càrrega de peticions (en viu/temps real) sobre els µserveis.
Condicionants	Cal aplicar-se quan hi haja un canvi de tendència (de baixa càrrega a alta, o viceversa).
Accions a aplicar	Aplicar el patró de "Load Balancer'", per a que incremente la capacitat computacional resultant o la reduïsca, en funció de si la càrrega de treball (símptoma) és alta o baixa, respectivament.

3.4.2. Requeriment d'adaptació: µserveis amb taxa de fallides alta

En aquest cas, passarem a descriure el requeriment d'adaptació per al *Circuit Breaker*. Aquest s'aplicarà seguint els criteris descrits a continuació, sempre i quan es complisquen certes condicions. La finalitat d'aquesta acció correctiva és simular que el µservei en qüestió continua disponible i funcional encara que no ho estiga.

Circuit Breaker	
Requeriment adaptació	Aplicar estratègia de control d'incidències.
Inputs d'informació o símptomes	Percentatge d'errades d'un µservei.
Condicionants	Cal aplicar-se quan es sobrepassen els llindars establerts.
Accions a aplicar	Aplicar el patró de "Circuit Breaker", per a que controle les errades del µservei i intervinga quan aquest supere el llindar d'errades establert.

3.4.3. Requeriment d'adaptació: un sol µservei per contenidor

En aquest cas, passarem a descriure el requeriment d'adaptació per mantenir un mètode de desplegament de µserveis. Aquest s'aplicarà seguint els criteris descrits a continuació, sempre i quan es complisquen certes condicions. La finalitat d'aquesta acció correctiva és mantenir un patró de desplegament de µserveis a la solució per poder tindre un millor control d'aquesta.

Service Per Container	
Requeriment adaptació	Aplicar estratègia de desplegament
Inputs d'informació o símptomes	Ubicació dels recursos.
Condicionants	Cal aplicar-se quan hi haja un canvi de tendència (de baixa càrrega a alta, o viceversa).
Accions a aplicar	Aplicar el patró de "Service Per Container", per a que reestructure la solució amb aquesta estratègia de desplegament.

3.4.4. Requeriment d'adaptació: repartiment de càrrega

En aquest cas, passarem a descriure el requeriment d'adaptació per al repartiment de càrrega. Aquest s'aplicarà seguint els criteris descrits a continuació, sempre i quan es complisquen certes condicions. La finalitat d'aquesta acció correctiva és disminuir la càrrega computacional que esta sofrint un *host* en un determinat moment.

Moure recursos	
Requeriment adaptació	Necessitat de moure recursos.
Inputs d'informació o símptomes	Nivell de càrrega computacional (en viu/temps real) sobre els *host*.
Condicionants	Cal aplicar-se quan un *host* tinga sobrecàrrega. Depenent de la quantitat, es moura un/varios µserveis/contenidors.
Accions a aplicar	Aplicar el patró de "Moure recursos", per a que reestructure la solució per reduir l'estrès dels *Hosts*.

3.4.5. Requeriment d'adaptació: recursos sempre disponibles i optimitzats

En aquest cas, passarem a descriure els requeriments d'adaptació per al control de la disponibilitat. Aquests s'aplicaran seguint els criteris descrits a continuació, sempre i quan es complisquen certes condicions. La finalitat d'aquestes accions correctives és assegurar que els µserveis que estiguen en ús estiguen sempre disponibles i els que no estiguen en ús estiguen replegats per optimitzar els recursos.

Copiar recursos	
Requeriment adaptació	Necessitat de copiar recursos.
Inputs d'informació o símptomes	Nivell de càrrega computacional (en viu/temps real) sobre els recursos.
Condicionants	Cal aplicar-se quan es dispare la adaptació de balanceig de càrrega per fer els clons que siguen necessaris.
Accions a aplicar	Aplicar el patró de "Copiar recursos", per a que proporcione la quantitat de còpies requerides.

Esborrar recursos	
Requeriment adaptació	Necessitat de esborrar recursos.
Inputs d'informació o símptomes	Nivell de carrega computacional null a un *host* o contenidor o petició d'esborrat.
Condicionants	Cal aplicar-se quan un *host* no tinga càrrega computacional durant una estona o quan es reba una petició d'esborrat d'un recurs.
Accions a aplicar	Aplicar el patró de "Esborrar recursos", per a esborrar el recurs que se'ns indique.

4. Disseny de µserveis autònoms

En aquest capítol exposarem el disseny de la solució a nivell arquitectònic. Per fer el disseny de la solució ens basarem en l'arquitectura del bucle de control MAPE-K; per tant, dividirem aquest capítol en dues seccions, una on comentarem com hem dissenyat el subsistema gestionat i una altra en què explicarem com hem dissenyat els diferents components del bucle MAPE-K i com els hem ubicat a la solució.

Començarem explicant com hem dissenyat el subsistema gestionat, que en aquest cas es tracta d'un conjunt de µserveis que formen part d'una solució de programari.

4.1. Subsistema gestionat

Per dissenyar el subsistema gestionat, ho farem com si fos qualsevol tipus de solució basada en µserveis, per tant el que necessitem és dividir les funcionalitats de les que consta la solució i desenvolupar-les com xicotets serveis on cadascun d'aquests realitza una funció determinada.

A més, necessitem afegir-li al subsistema una sèrie de sensors i efectors per a que es puga comunicar amb el bucle de control. D'aquesta manera mitjançant els sensors/sondes, obtindrem les dades necessàries per a avaluar la situació i decidir si és necessari aplicar una adaptació i mitjançant els efectors, poder realitzar els canvis necessaris a la solució per poder aplicar correctament el procés d'adaptació.

4.1.1. Sensors

Els sensors recullen les dades sobre la solució i les reporten al bucle de control, en concret als monitors. Són peces que solen estar incrustades dins del sistema gestionat i són responsables de mesurar o obtenir indicadors de funcionament (o errades) per a reportar al bucle de control a través dels monitors.

Com que el que estem buscant és construir una solució de µserveis que s'autoregule, necessitem sondes que en reporten quan puguen haver indicadors de funcionament que ens conduïsquen a potencials situacions en què el bucle de control haja de prendre alguna mesura.

A més, com que les accions de regulació estan realitzades pels patrons identificats, necessitem mesurar indicadors de funcionament dirigits cap a l'aplicació d'aquests patrons. Així que, per a cada patró, haurem d'identificar la sonda o les sondes que ens reporten la informació que siga rellevant en l'àmbit d'aplicació del patró.

Per tant, les sondes que utilitzarem seran les següents:

- **Sonda per al patró "Load Balancer":** Necessitarem mesurar la càrrega de peticions que està rebent qualsevol µservei de la solució desplegada. Per a no estar reportant mesures per a tots els valors possibles, establirem uns llindars superiors i inferiors de manera que la sonda reportarà si està mesurant per damunt (*high*) o per baix (*low*) d'aquests llindars. Aquesta sonda haurà de reportar les seues mesures al **monitor "Load Balancer"**, definit a la propera secció.

- **Sonda per al patró "Circuit Breaker":** Necessitarem mesurar els errors que cometen els µserveis, com per exemple errors de comunicació. Per no estar reportant valors cada vegada que ocorre un error, establirem uns llindars superiors i inferiors i establirem un percentatge d'error. Si aquest percentatge supera el llindar establert, aquesta sonda reportarà les mesures al **monitor "Circuit Breaker"**, definit a la propera secció.

- **Sonda per al patró "Service Per Container":** Necessitarem mesurar la quantitat de quantitat de µserveis als contenidors i als *hosts* per comprovar que hi ha zero µserveis als *hosts* i un µservei a cada contenidor, de manera que confirmem que tots els µserveis estan encapsulats a contenidors i sols hi ha un per contenidor. Reportarem els valors cada vegada que es detecte que un µservei no està col·locat adequadament. Si detectem aquesta situació, la sonda reportarà les mesures al **monitor "Service per Container"**, definit a la propera secció.

- **Sonda per a la funció "Moure recursos":** Necessitarem mesurar la càrrega computacional dels *hosts* per determinar si és necessari moure algun recurs a un altre *host*. Reportarem els valors cada vegada que un *host* arribe el llindar establert. Una vegada assolit el llindar, la sonda reportarà les mesures al **monitor "Moure recursos"**, definit a la propera secció.

- **Sonda per a la funció "Copiar recursos":** Necessitarem mesurar la càrrega de peticions que està rebent qualsevol µservei de la solució desplegada. Per reportar els valors, fixarem uns llindars de manera que si s'assoleixen, es copien els recursos necessaris per a que es puga aplicar el patró "Load Balancer". Una vegada aconseguit el llindar, la sonda reportarà les mesures al **monitor "Copiar recursos"**, definit a la propera secció.

- **Sonda per a la funció "Esborrar recursos":** Necessitarem mesurar la càrrega computacional dels *hosts* per determinar si els *hosts* estan en ús. Per reportar els valors, fixarem uns llinars de manera que si s'assoleixen i es mantenen durant un determinat temps, s'esborraran els recursos degut a que no estan en ús. Una vegada aconseguit el llindar, la sonda reportarà les mesures al **monitor "Esborrar recursos"**, definit a la propera secció.

4.1.2. Efectors

Aquesta capa és l'encarregada de rebre les peticions d'adaptació sobre recursos de la solució, i provocar els canvis pertinents en la infraestructura d'execució. Així, haurà d'oferir una interfície que puguen emprar les peces del bucle de control per a executar aquestes adaptacions.

La capa d'efectors també serveix com a pivot per a desacoblar als elements del bucle de control de detalls i restriccions tecnològiques imposades per la implementació del sistema gestionat. Aquesta capa ha d'oferir una interfície coneguda amb anterioritat pels elements del bucle de control, de manera que puguen sol·licitar aquestes operacions de canvi sobre la solució. D'altra banda, ha de proposar una implementació concreta d'aquestes operacions que siguen adients amb la tecnologia amb la que està desenvolupat el sistema gestionat.

En l'àmbit de la solució de µserveis, aquesta capa d'efectors està dividida en tres grups d'interfícies i implementacions, associats als recursos principals: nodes, contenidors i µserveis.

4.2. Bucle de control MAPE-K

Per fer el disseny del bucle de control utilitzarem el bucle MAPE-K proposat per IBM. Aquest bucle consta de sis elements comentats anteriorment on ubicarem els components que descriurem a continuació. Gràcies a aquests components podrem aconseguir que el subsistema gestionat i el bucle MAPE-K interactuen entre si de manera correcta.

Els components en qüestió són les sondes, els diferents mòduls (Monitorització, Anàlisi, Planificació, Execució i Coneixement) i les peces que els constitueixen (monitors, propietats d'adaptació, regles d'adaptació, planificador i executor). S'ubiquen al bucle MAPE-K de la següent manera:

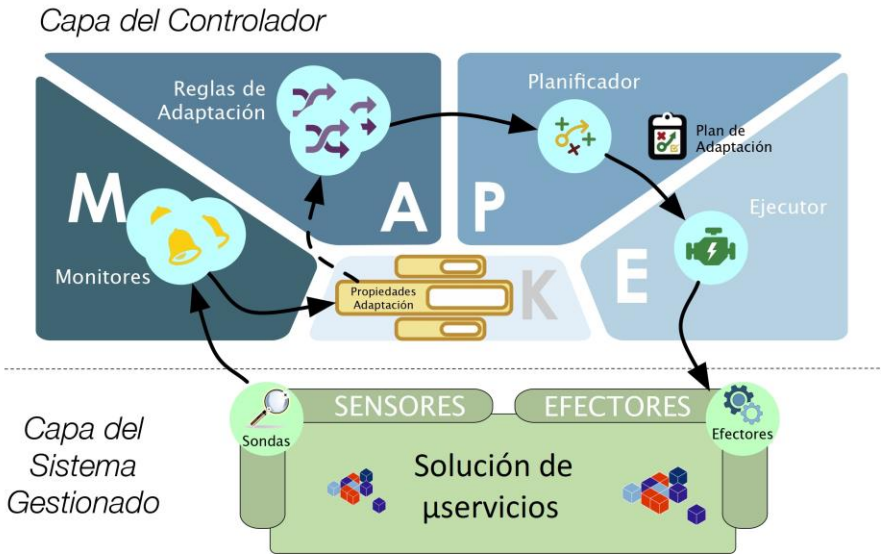

Figura 22: Disseny de µserveis autònoms.

Com podem veure a la figura 22, hem distribuït els diferents components que hem desenvolupat de manera que ens quede el bucle MAPE-K preparat per poder funcionar adequadament. En la nostra solució emprarem el *framework* FADA per a solucions amb µserveis, que ja ens proporciona un disseny de components per a totes les peces del bucle, i a més, una implementació per a tots els mòduls, per al planificador i per a l'executor.

Aleshores, en aquesta tesina, sols hem de centrar-nos en descriure (i desenvolupar) els següents tipus d'elements per a completar el bucle: monitors, propietats d'adaptació i regles d'adaptació. Les properes seccions presenten els components identificats per a satisfer les necessitats establertes.

4.2.1. Monitors i propietats d'adaptació

Els monitors són els encarregats de rebre les mesures enviades per les sondes, filtrar-les, acumular-les, integrar-les i/o transformar-les en informació d'utilitat de cara a identificar símptomes en el sistema que està sent gestionat. És responsabilitat de cada monitor entendre el tipus de mesura que està rebent per a poder manipular-la i gestionar-la adequadament. Allò més important és el procés de conversió de la dada "en brut" en informació rellevant o símptoma. Aquests símptomes es reporten com a propietats d'adaptació en el bucle de control, i s'emmagatzema en el mòdul de Coneixement (*Knowledge*) del propi bucle.

Per a atendre a les sondes identificades a la secció anterior, caldran els següents monitors i propietats d'adaptació:

- **Monitor de càrrega de peticions d'un µservei** : Rebrà mesures sobre la càrrega de treball que està atenent un µservei. El seu objectiu és identificar un símptoma de que s'està "estressant molt" o "relaxant amb excés" la solució actual, i provocar que es dispare una regla d'adaptació que tracte de ficar més recursos disponibles o replegar-los, si ja no calen. Aquest monitor es recolzarà sobre un **propietat d'adaptació** "**càrrega de peticions**" per a un µservei concret. Aleshores, si rep una lectura d'alta càrrega de treball, i la seua càrrega de peticions anterior era baixeta o no hi havia informació encara, actualitzarà aquesta càrrega de peticions per al µservei indicat indicant que és ara "alta". De la mateixa manera, si es rep una mesura de baixa càrrega de peticions sobre un µservei i la seua anterior situació era "alta" o "no definida", s'actualitzarà aquest valor amb el nou valor "baix".

- **Monitor de taxa d'errors d'un µservei:** Rebrà mesures sobre la taxa d'errors que està tenint un µservei. El seu objectiu és identificar un símptoma de que està funcionant malament el µservei en qüestió i provocar que es dispare una regla d'adaptació que desplegue un servei de tipus "Circuit breaker" per previndre els errors o la indisponiblitat del µservei afectat. Aquest monitor es recolzarà sobre un **propietat d'adaptació** "**taxa d'errors**" per a un µservei concret. D'aquesta manera, sempre que un µservei supere el llindar, se li aplicarà aquest patró com a mesura de prevenció. A més, si en un determinat moment la taxa de falles baixa a mínims, el llindar inferior ho detectarà i retirarà el component "Circuit Breaker".

- **Monitor de control del desplegament:** Rebrà mesures sobre la ubicació dels µserveis. El seu objectiu és controlar que el desplegament segueix un patró en tot moment de manera que, si no es compleix, es dispararà una regla d'adaptació que repare la situació. Aquest monitor es recolzarà sobre un **propietat d'adaptació** "**control del desplegament**" per a un µservei concret. D'aquesta manera sempre que un µservei es trobe a una posició incorrecta (desplegat a un *host* sense estar a dintre d'un contenidor o a dintre d'un contenidor on ja hi havia un µservei) el replegarà per tornar-lo a desplegar correctament.

- **Monitor de copiat de recursos:** Rebrà mesures sobre la necessitat d'incloure recursos. El seu objectiu és desplegar còpies dels recursos sempre que siga necessari, de manera que, si arriba una petició de còpia es dispare una regla d'adaptació que realitze la quantitat de còpies indicades a la mesura. Aquest monitor es recolzarà sobre un **propietat d'adaptació** "**clonar recurs**" per a un µservei concret. D'aquesta manera, sempre que un µservei necessite ser copiat, es realitzaran tantes còpies d'aquest com indique la mesura i es desplegaran a la solució on siguen necessaris.

- **Monitor de moviment de recursos:** Rebrà mesures sobre la càrrega computacional dels *hosts*. El seu objectiu és controlar el nivell de càrrega computacional de manera que si aquest supera uns llindars, es llance una regla d'adaptació per moure recursos a un altre *host* menys carregat. Aquest monitor es recolzarà sobre un **propietat d'adaptació** (símptoma) "**moure recurs**" per a un µservei concret. D'aquesta manera sempre que un *host* es trobe saturat, podrem llançar una adaptació per moure recursos a una altra ubicació i disminuir la càrrega computacional d'aquest.

- **Monitor d'esborrat de recursos:** Rebrà mesures sobre la càrrega computacional dels recursos. El seu objectiu és controlar els recursos estan en ús, si l'ús d'un recurs és nul durant una estona, podem suposar que no s'utilitza i, per tant, podem esborrar-lo per alliberar espai i capacitat de còmput. Aquest monitor es recolzarà sobre un **propietat d'adaptació** "**esborrar recurs**" per a un µservei concret. D'aquesta manera, sempre que un recurs no estiga en ús i detectem que no s'utilitza, llançarem una regla d'adaptació per esborrar-lo ja que es suposa que no és necessari.

4.2.2. Regles d'adaptació

Les regles d'adaptació són les responsables de definir l'adaptació que s'ha d'aplicar al sistema que s'està gestionant per a regular-lo quan esdevé algun símptoma, o un conjunt de símptomes. A les regles d'adaptació es defineixen com s'ha de quedar aquest si la regla s'executa. Les regles han d'atendre a canvis en els símptomes o propietats d'adaptació, sobre els que es subscriuen.

D'aquesta manera, seguint la proposta que s'està fent, es requereixen les següents regles:

- **Regla d'adaptació per a canvis en la càrrega de treball/peticions d'un µservei:** Es subscriuran a canvis en totes les propietats d'adaptació que controlen la càrrega de peticions per a cada µservei. En cas que s'active la regla, haurà de veure si s'està sofrint una situació de sobrecàrrega de treball o de relaxació. Segons siga, haurà d'aplicar una o altra acció correctiva, segons s'ha descrit el patró a la secció 3.2.1.

- **Regla d'adaptació per a canvis en la disponibilitat d'un µservei:** Es subscriuran a canvis en totes les propietats d'adaptació que controlen la taxa d'errors dels µserveis. En cas que s'active la regla, haurà de veure si s'està sofrint una situació de pèrdua de disponibilitat. Depenent del cas, haurà l'acció correctiva, segons s'ha descrit el patró a la secció 3.2.2.

- **Regla d'adaptació per al control del desplegament:** Es subscriuran a canvis en totes les propietats d'adaptació que controlen la ubicació dels µserveis. En cas que s'active la regla, haurà de veure si el µservei està sense encapsular o encapsulat amb un altre µservei. Depenent del cas, haurà l'acció correctiva, segons s'ha descrit el patró a la secció 3.2.3.

- **Regla d'adaptació per clonar un servei µserveis:** Es subscriuran a canvis en totes les propietats d'adaptació que controlen la càrrega computacional dels µserveis. En cas que s'active la regla, haurà de veure quin llindar s'està sobrepassant. Depenent del cas, haurà l'acció correctiva, segons s'ha descrit el patró a la secció 3.3.1.

- **Regla d'adaptació per esborrar µserveis:** Es subscriuran a canvis en totes les propietats d'adaptació que controlen la càrrega computacional dels µserveis. En cas que s'active la regla, haurà de veure quan de temps porta el µservei sense funcionar. Depenent del cas, haurà l'acció correctiva, segons s'ha descrit el patró a la secció 3.3.2.

- **Regla d'adaptació per a moure d'un µservei:** Es subscriuran a canvis en totes les propietats d'adaptació que controlen la càrrega computacional dels µserveis. En cas que s'active la regla, haurà de veure quin llindar s'ha sobrepassat per veure quants µserveis s'han de moure. Depenent del cas, hi haurà l'acció correctiva, segons s'ha descrit el patró a la secció 3.3.3.

5. Implementació de µserveis autònoms

En aquest capítol, comentarem més en detall com hem fet la implementació dels components de la solució, tant del subsistema gestionat com dels components del bucle MAPE-K. Per tant, explicarem com hem desenvolupat el subsistema i com hem desenvolupat els components del bucle MAPE-K que es mostren a la següent figura:

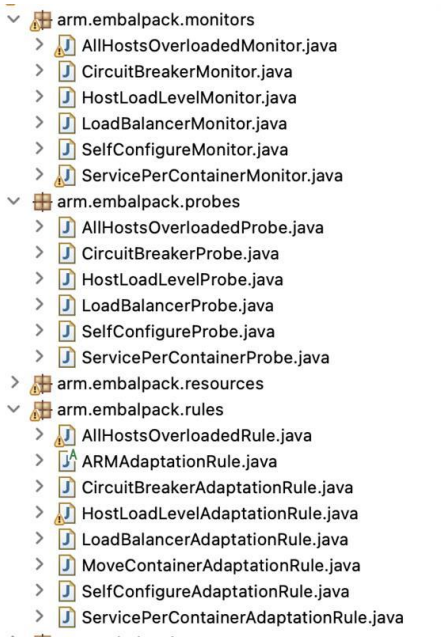

Figura 23: Components del bucle MAPE-K.

5.1. Implementació del subsistema gestionat

Com hem vist al capítol de disseny, s'han d'implementar una sèrie de components per poder fer un ús correcte del sistema dissenyat. Per a explicar com hem fet la implementació utilitzarem d'exemple el patró *Load Balancer*.

Per fer la implementació del subsistema gestionat, hem utilitzat unes llibreries que s'han desenvolupat al PRoS i que ens permeten simular µserveis. Gràcies a aquestes llibreries hem aconseguit fer una solució basada en µserveis que simula l'escenari d'Embalpack descrit anteriorment.

A més, hem utilitzat les llibreries per poder afegir els sensors i efectors necessaris per a que el bucle de control puga interactuar amb el subsistema i així aplicar-li les adaptacions que siguen necessàries. Aquesta llibreria ens ofereix una implementació d'una capa de μserveis amb OSGi, pel que no necessitem implementar-la. Per a poder dur a terme el treball presentat a la tesina, només caldrà que implementem la capa de sensors.

5.1.1. Implementació dels sensors

Per implementar els sensors, hem de realitzar dues funcions, una que ens cree la sonda i una altra que realitze l'acció que li indiquem, en aquest cas, enviar una sol·licitud d'adaptació al monitor, ja que s'ha detectat una situació anòmala que ha de ser reportada. A continuació es comentaran aquestes funcions amb més detall:

```
12  public class LoadBalancerProbe extends ARMSimulationProbe {
13
14      public LoadBalancerProbe(BundleContext context) {
15          super(context, "loadbalancer-configure-probe");
16      }
```

Figura 24: Generació del sensor.

Com es pot apreciar, a la primera funció el que fem és crear un nou sensor on li indiquem que aquest serà un "loadbalancer-configure-probe" (línia 15). D'aquesta manera, mitjançant les llibreries que tenim disponibles (línia 12), crearem cadascun dels sensors que hem comentat a l'inici del capítol. La llibreria per crear els sensors, està dissenyada per a que sols siga necessari enviar-li l'entorn i el tipus de sensor. Per facilitar la feina del programador, les llibreries per a desenvolupar els sensors, monitors i regles d'adaptació estan dissenyades de la mateixa manera, de manera que realitzant canvis mínims relatius a la funcionalitat i l'objectiu del component, serem capaços de desenvolupar qualsevol d'aquests components.

```
18      public void sendLoadBalancerConfigureRequest(ARMMeasure theMeasure) {
19          this.reportMeasure(theMeasure);
20      }
```

Figura 25: Funció del sensor.

Aquesta funció, es l'encarregada d'enviar les peticions d'adaptació al monitor, de manera que cada vegada que el sensor detecta una situació anòmala, s'envia una petició d'adaptació per a que es comprove i es repare el més prompte possible (línia 18).

5.2. Implementació dels components dels bucle MAPE-K

Com s'ha vist al capítol de disseny, necessitem una sèrie de components per a que el bucle de control puga funcionar correctament, com per exemple monitors o regles d'adaptació. Per explicar com hem fet el desenvolupament utilitzarem el mateix exemple que hem utilitzat per explicar com s'implementarien els components del subsistema gestionat, és a dir, el patró *Load Balancer*.

Per a implementar el bucle utilitzarem també una sèrie de llibreries que ha proporcionat el tutor i ens permeten crear els components del bucle d'una manera bastant simple i efectiva. Necessitarem implementar, monitors, sondes i regles d'adaptació que després utilitzarem per fer les possibles adaptacions i així aconseguir que el sistema s'autogestione.

Per fer la implementació, necessitarem desenvolupar tants monitors, sondes, regles, etc. com hem definit a l'anàlisi i hem dissenyat al capítol de disseny, de manera que cada patró o funció obtinga els seus pròpies components del bucle MAPE-K.

Donat que aquests components es desenvolupen d'una manera semblant independentment de la seua finalitat (és igual que siga per al *Circuit Breaker* o per al *Load Balancer*, els dos es desenvolupen igual) utilitzarem els components del *Load Balancer* per mostrar la implementació de cadascun dels components. A continuació es mostrarà com s'han implementat aquests components que conformen la solució.

5.2.1. Implementació dels monitors

Per implementar el monitor, hem realitzar dues funcions, una que ens crea el monitor i una altra que realitza l'acció que li hem indicat, en aquest cas, emmagatzemar les dades de tipus "loadbalancer-configure" a la base de dades del coneixement. A continuació es comentaran aquestes funcions amb més detall:

```
11  public class LoadBalancerMonitor extends Monitor {
12
13      public LoadBalancerMonitor(BundleContext context) {
14          super(context, "loadbalancer-monitor");
15      }
```

Figura 26: Generació del monitor.

Com es pot apreciar, a la primera funció el que fem és crear un monitor nou on li indiquem que aquest serà un "loadbalancer-configure-monitor" (línia 14). Podem veure que, com ja s'ha comentat abans, aquesta funció segueix la mateixa estètica que la dels sensors gracies a l'ús de les llibreries. Després, utilitzant les llibreries, crearem cadascun dels monitors que hem comentat a l'inici del capítol (línia 13). Per crear els monitors, la llibreria està dissenyada per a que sols siga necessari enviar-li l'entorn i el tipus del monitor.

```
18      public IMonitor report(Object measure) {
19
20          logger.debug(String.format("(Load Balancer Monitor) Received measure: %s", measure.toString()));
21
22          try {
23              ARMMeasure theMeasure = (ARMMeasure)measure;
24              IKnowledge knowledge = this.getTheMonitoringModule().getTheKnowledgeModule().getTheKnowledge();
25              String knowledgePropertyId = "current-incoming-requests-"+theMeasure.getThing();
26              IKnowledgeProperty kp = knowledge.getKnowledgeProperty(knowledgePropertyId);
27              if ( kp == null )
28                  kp = knowledge.createKnowledgeProperty(knowledgePropertyId, measure);
29              else
30                  kp.setValue(measure);
31          } catch (Exception e) {
32              return this;
33          }
34
35          return this;
36      }
```

Figura 27: Funció del monitor.

Aquesta funció rebrà qualsevol dada que li arribe i comprovarà que és del tipus corresponent al monitor, en aquest cas "loadbalancer-configure", si la dada és d'aquest tipus, l'emmagatzemarà a la base de dades del coneixement per a que posteriorment se li aplique l'adaptació que siga necessària.

5.2.2. Implementació de les regles d'adaptació

Per implementar la regla d'adaptació hem seguit la mateixa estratègia que amb la implementació dels anteriors, crearem una funció per a construir la regla d'adaptació utilitzant la llibreria en qüestió (LoadBalancerConfigureAdaptationRule extends ARMAdaptationRule) i una altra per a que execute les accions que siguen necessàries.

Per a la regla d'adaptació, a banda del que ja hem vist anteriorment, entren en joc dues variables noves: "theCurrentSystemConfiguration", "theNextSystemConfiguration". Aquestes variables contenen la configuració de la solució actual i la configuració que s'aplicarà a continuació. Per tant, per evitar possibles errors en dependències o ubicació d'elements, totes les modificacions que realitzarem en una iteració, es faran a "theNextSystemConfiguration" i una vegada acabades, enviarem aquesta variable al sistema per a que actualitze la variable (i la solució) "theCurrentSystemConfiguration" i així obtindre un modelat estable.

```
17  public class LoadBalancerAdaptationRule extends ARMAdaptationRule {
18
19      public LoadBalancerAdaptationRule(BundleContext context) {
20          super(context, "LOAD-BALANCER");
21          this.setListenToKnowledgePropertyChanges("current-incoming-requests-*");
22      }
23
```

Figura 28: Generació de la regla.

A la primera funció (línia 19), crearem la regla d'adaptació (de tipus "loadbalancer-configure-rule") i afegirem un listener (línia 21) per a controlar els canvis que puguen ocórrer a la base de dades del coneixement respecte a la propietat "loadbalancer-configure". De manera que en el moment en què aquesta canvie, la regla d'adaptació s'executarà i farà les accions necessàries per a realitzar l'adaptació.

```
62  public IRuleSystemConfiguration onExecute(IKnowledgeProperty property) throws RuleException {
63
64      String microserviceType = null;
65      String measure = null;
66
67      try {
68
69          microserviceType = (String)theMeasure.getThing();
70          measure = (String) theMeasure.getMeasure();
71
72          String previousKpId = "previous-incoming-requests-"+theMeasure.getThing();
73          IKnowledgeProperty kp = this.getTheKnowledge().getKnowledgeProperty(previousKpId);
74          if ( kp == null )
75              this.getTheKnowledge().createKnowledgeProperty(previousKpId, measure);
76          else
77              kp.setValue(measure);
78
79
80
81      } catch (Exception e) {
82          logger.trace("Cannot understand knowledge property value. Not executing the rule ...");
83          throw new RuleException("Unknown knowledge property value", "Cannot understand knowledge property value. Not executing the rule ...");
84      }
85
86      IRuleMicroservicesSystemConfiguration theNextSystemConfiguration = RuleMicroservicesSystemConfiguration.build(this.getId() + "_" +
87       ITimeStamped.getCurrentTimeStamp(), REFERENCE_MODEL);
88      IMicroservicesSystemConfiguration theCurrentSystemConfiguration = (IMicroservicesSystemConfiguration) this.getTheAnalyzingModule().getTheKnowledgeModule().
89      getTheKnowledge().getCurrentSystemConfiguration();
90
91      MicroservicesPatternsHelper.setBundleContext(context);
92
93      if ( measure.equalsIgnoreCase("very-high") )
94          return MicroservicesPatternsHelper.applyLoadBalancer(theNextSystemConfiguration, theCurrentSystemConfiguration, microserviceType, 6);
95      if ( measure.equalsIgnoreCase("high") )
96          return MicroservicesPatternsHelper.applyLoadBalancer(theNextSystemConfiguration, theCurrentSystemConfiguration, microserviceType, 3);
97      if ( measure.equalsIgnoreCase("low") ) {
98          return MicroservicesPatternsHelper.unrollLoadBalancer(theNextSystemConfiguration, theCurrentSystemConfiguration, microserviceType, 2);
99      }
100
101     return theNextSystemConfiguration;
102 }
103
```

Figura 29: Funcions de la regla.

Com es pot apreciar, aquesta funció es podria dividir en cinc parts.

A la primera part (línies 67-85), comprovem que les dades de la mostra que hem rebut (perquè algú ha realitzat un canvi a la base de dades del coneixement) són vàlides i es poden utilitzar. Per fer açò, hem controlat diverses excepcions que es podrien produir i així ens assegurem que la mostra que utilitzarem és correcta.

A la segona part (línies 86-87) ens creem la variable per a poder guardar la configuració de l'adaptació que realitzarem i que en un futur es convertirà en el nou modelat de la solució.

A la tercera part (línia 88-89) ens creem la variable per poder guardar la configuració del modelat actual i utilitzar-la per fer l'adaptació corresponent. Per obtindre el valor que estem buscant, sols necessitem consultar la base de dades del coneixement i veure quina és l'última configuració disponible.

A la quarta part (línia 91) establim el context de l'adaptació per poder treballar amb el modelat de manera adequada coneixent quin és l'entorn.

I, finalment, a la quinta part (93-99) és on utilitzarem aquesta funció per indicar les accions que s'han de realitzar per aplicar el patró corresponent, en aquest cas un Load Balancer. Per a aplicar aquest patró en concret, necessitem tant la configuració actual com la següent i un nombre de còpies que ve determinat per la càrrega computacional del µservei al qual es pretén aplicar-li el balancejador de càrrega. A continuació veurem un extracte de la funció "applyLoadBalancer" on comentarem com funcionen aquestes funcions ubicades a la llibreria auxiliar que hem desenvolupat anomenada "MicroservicesPatternsHelper".

```
178         int i = 1;
179         while( i <= num_clones ) {
180 
181             newMicroservice = MicroservicesBuilder.copyMicroservice(sourceMicroservice);
182             id = newMicroservice.getInstanceId();
183 
184             newHost = mostUnloadedHosts.get(hostAllocations++);
185             HostsBuilder.copyEndpoints(sourceHost, sourceContainer.getReference(), newHost);
186 
187             newContainer = ContainersBuilder.copyContainer(sourceContainer, id, newHost.getReference());
188             ContainersBuilder.copyEndpoints(sourceContainer, sourceMicroservice.getReference(), newContainer);
189             HostsBuilder.copyEndpointDelegations(sourceHost, sourceContainer.getReference(), newHost, newContainer.getReference());
190 
191             newMicroservice.setContainer(newContainer.getReference());
192             MicroservicesBuilder.copyServiceSupplies(sourceMicroservice, newMicroservice);
193             MicroservicesBuilder.copyServiceRequires(sourceMicroservice, newMicroservice);
194             MicroservicesBuilder.copyBindings(sourceMicroservice, newMicroservice);
195 
196 
197             ContainersBuilder.copyEndpointDelegations(sourceContainer, sourceMicroservice.getReference(), newContainer, newMicroservice.getReference());
198 
199             if ( newMicroservice_lb.getSuppliedServices() != null ) {
200                 for(IMicroserviceEndpoint endpoint : newMicroservice_lb.getSuppliedServices().values() ) {
201 
202                     suppliedEndpoint = sourceMicroservice.getSuppliedService(endpoint.getEndpointName());
203                     IContainerEndpointDelegation cDelegation = MicroservicesBuilderUtils.searchContainerEndpointDelegation(suppliedEndpoint);
204                     IHostEndpointDelegation hDelegation = MicroservicesBuilderUtils.searchHostEndpointDelegation(cDelegation.getContainerEndpointReference());
205                     if ( hDelegation != null ) {
206                         IBindingSpecification binding = new BindingSpecificationBuilder(context).
207                                 setRequiredMicroserviceEndpointReference(newMicroservice_lb, "req"+endpoint.getEndpointName()).
208                                 publicBind(newHost.getReference(), hDelegation.getHostEndpointReference().getEndpointName()).
209                                 build();
210                         newMicroservice_lb.addBinding(binding);
211                     }
212 
213                 }
214             }
215 
216             HostsBuilder.addHostToNextSystemConfiguration(newHost, theNextSystemConfiguration);
217             ContainersBuilder.addContainerToNextSystemConfiguration(newContainer, theNextSystemConfiguration);
218             MicroservicesBuilder.addMicroserviceToNextSystemConfiguration(newMicroservice, theNextSystemConfiguration);
219 
220             i++;
221 
222         }
223 
```

Figura 30: Bucle de la funció "Apply Load Balancer".

Aquest extracte ens mostra el bucle que utilitzarem per crear/clonar tants µserveis com siga necessari per a poder desplegar el balancejador de càrrega. Com podem apreciar a l'extracte, necessitem copiar tots els elements del µservei a clonar, des dels *endpoint* (línies 185 i 188), fins als *bindings* (línia 194) o les delegacions (línia 197).

A més per al nou µservei (el que farà la funció de *Load Balancer*), haurem de crear tots els elements necessaris per a que puga utilitzar els elements copiats mencionats anteriorment (línies 199-214).

Finalment, sols necessitarem afegir els canvis a la configuració per a que quan acabe de fer l'adaptació aplique els canvis correctament i arribem a un estat desitjat (línies 216-218). A més cap destacar que durant tot el procés ens hem dedicat a modificar models ja que com hem mencionat anteriorment, les adaptacions es realitzen modificant el modelat de la solució i fent generació automàtica de codi a partir d'aquest. Així podem modificar el modelat i, mitjançant la generació de codi a partir del modelat, generar les adaptacions necessàries per a que el sistema s'autogestione correctament i s'adapte depenent de les seues necessitats realitzant xicotets ajustos al modelat de la solució.

Una vegada hem acabat el procés d'adaptació, indiquem el resultat a la variable "theNextSystemConfiguration" on s'inclourà el nou modelat després de realitzar l'adaptació.

6. Aplicació industrial pràctica

Per a definir l'escenari industrial utilitzarem el de la fàbrica Embalpack on recentment s'ha actualitzat a una arquitectura amb µserveis. El que farem serà dotar a aquesta infraestructura de computació autònoma, de manera que integrarem µserveis amb computació autònoma i veurem l'impacte que pot tindre aquest tipus de solucions en un entorn real com és el d'Embalpack.

El primer pas abans de veure com dissenyem la solució és veure quins són els recursos dels que disposem, per poder fer una anàlisi prèvia del que tenim i del que anem a necessitar per a desenvolupar la solució.

6.1. Cas d'estudi Embalpack

A la fàbrica d'Embalpack disposem de la següent infraestructura. Aquesta està capacitada per a comunicar-se com si es tractara d'un dispositiu IoT.

- **Recursos:**
    - **Bovines de paper:** Amb capacitat de comunicació amb l'entorn. Hi ha serveis que permeten emmagatzemar la seua informació, o altres que atenen a la seua localització.
    - **Operaris:** També generen informació de la seua participació en la màquina o en l'operació que estan realitzant.
    - **Carretó Elevador:** Emprades per dur les bovines d'una ubicació a l'altra.
    - **Equips:** Tenim sis equips encarregats de gestionar les cadenes de producció (quatre per a les màquines cantoneres i dos per a les màquines rebobinadores) i un equip per al ERP.
    - **Controladores:** Quatre controladores/equips de baix rendiment, encarregats de gestionar els semàfors de les cadenes de producció.

- **Màquines:**
  - **Màquines rebobinadores:** rebobinen bobines de paper en bobines més menudes. Atenen ordres de producció (que venen des de l'ERP de l'empresa), i reporten en temps real (plataforma IoT) el seu estat.
  - **Màquines que fan cantoneres:** Produeixen cantoneres a partir de bobines generades per les rebobinadores. També generen informació sobre la producció, atenent a les ordres de producció generades per l'ERP.

- **(micro) Serveis:**
  - **Gestió d'estoc:** Controla l'estoc de bobines i cantoneres.
  - **Gestió de recursos:** Controla els recursos disponibles.
  - **Gestió de localitzacions:** Controla la ubicació dels recursos.

- **ERP de producció:** Aquest no és un µservei, i potser siga complicat argumentar que ho és. Però, li hem desenvolupat infraestructura per a que es puga connectar a la plataforma IoT a través d'una API REST i comunicar-se mitjançant MQTT amb diferents recursos i màquines.

A més, com es pretén treballar seguint l'estètica dels dispositius IoT, on cada dispositiu compta amb la seua versió digital on emmagatzema el seu estat actual, s'ha decidit incloure una sèrie de µserveis extra per controlar els diversos aspectes dels recursos i així augmentar la funcionalitat del sistema i apropar-lo més al que seria una típica infraestructura d'IoT:

- **Gestió del registre:** Registra tots els objectes digitals dels que disposem.
- **Objectes digitals:** Emmagatzema l'estat del objecte digitalitzat.
- **Registre de les transformacions:** Registra les transformacions que sofreixen les bovines, produïdes per les màquines.

Una vegada tenim enumerada la infraestructura de què disposem, passarem a veure com interactuen els dispositius entre ells i ho detallarem tot a un esquema lògic de la solució.

6.2. Aplicació de µserveis a Embalpack

Tenint en compte els recursos de la fàbrica, dissenyarem un esquema lògic on detallarem amb qui es comunicarà cada µservei i, a més, adquirirem una vista generalitzada del que seria la solució que proposem per a la fàbrica d'Embalpack.

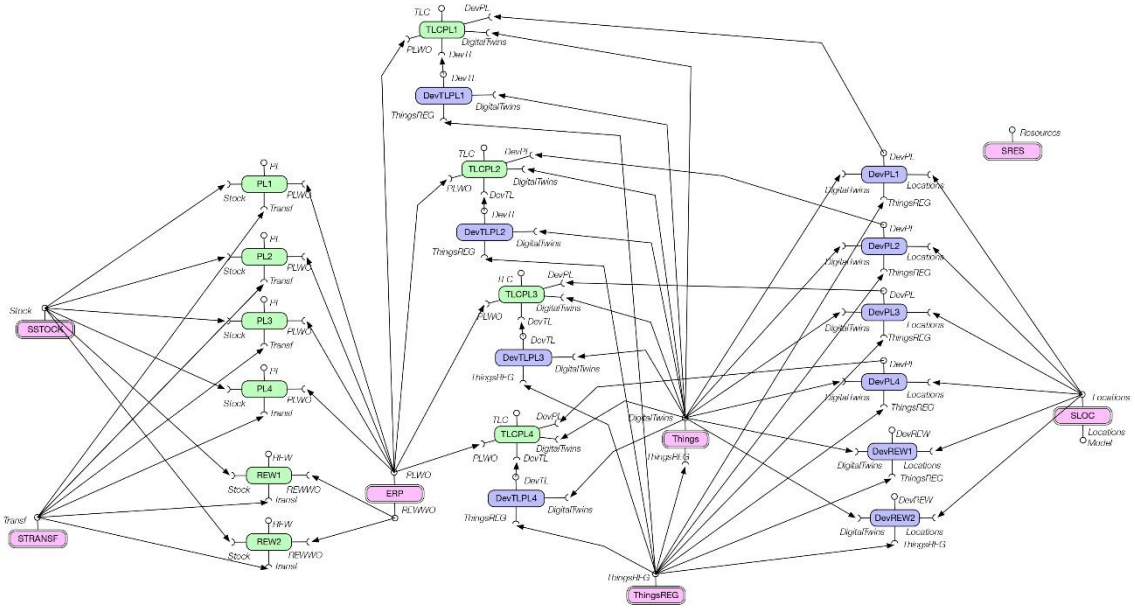

Figura 31: Esquema lògic de la solució industrial.

Després de dissenyar l'esquema lògic, podem tindre una visió mes clara de les dependències entre µserveis i com hauríem de fer el desenvolupament d'aquests. Per tant, amb la infraestructura que tenim i els tres µserveis que hem afegit per a controlar l'estat actual dels recursos, l'esquema lògic ens quedaria de la següent manera:

A banda de l'esquema lògic, hem detallat a l'annex els diferents components de l'esquema, on comentem per a cada component les interfícies que requereix o proporciona, la funcionalitat que aporta, les instàncies que desplega en un primer moment i les connexions que ha de tindre per a que funcione correctament.

Una vegada dissenyat l'esquema lògic, procedirem a veure com implementaríem la solució. En aquest cas, la solució base ja estava implementada i ens la proporciona el tutor del projecte, per tant nosaltres sols ens hem de preocupar de fer el desplegament de la solució, aconseguir adaptar-la per a que treballe amb bucles de control i adaptar els patrons i funcionalitats que hem implementat i explicat en capítols anteriors a aquesta solució.

Començarem per detallar un desplegament inicial i a partir d'aquest punt integrar els bucles de control i adaptar els patrons.

6.3. Desplegament inicial de referència

Per distribuir els components als dispositius disponibles hem de tindre en compte les restriccions de la solució. Hi ha µserveis que han d'estar en una màquina en concret per poder funcionar, aquestes restriccions estan detallades a l'annex; per tant, després de consultar-les, procedirem a realitzar un possible desplegament de la solució.

Degut a les restriccions hem pensat que la distribució més eficient per a un desplegament inicial on les màquines i dispositius no presenten càrrega computacional, seria distribuir els µserveis depenent de la seua funcionalitat, és a dir, si el µserveis PL1 treballa a la màquina de la cadena de producció nº1, desplegar-lo a aquesta màquina. Per altra banda, els µserveis amb funcions generals, per exemple el µservei de localització, que s'utilitzen a diversos punts, es desplegarien a la màquina del ERP.

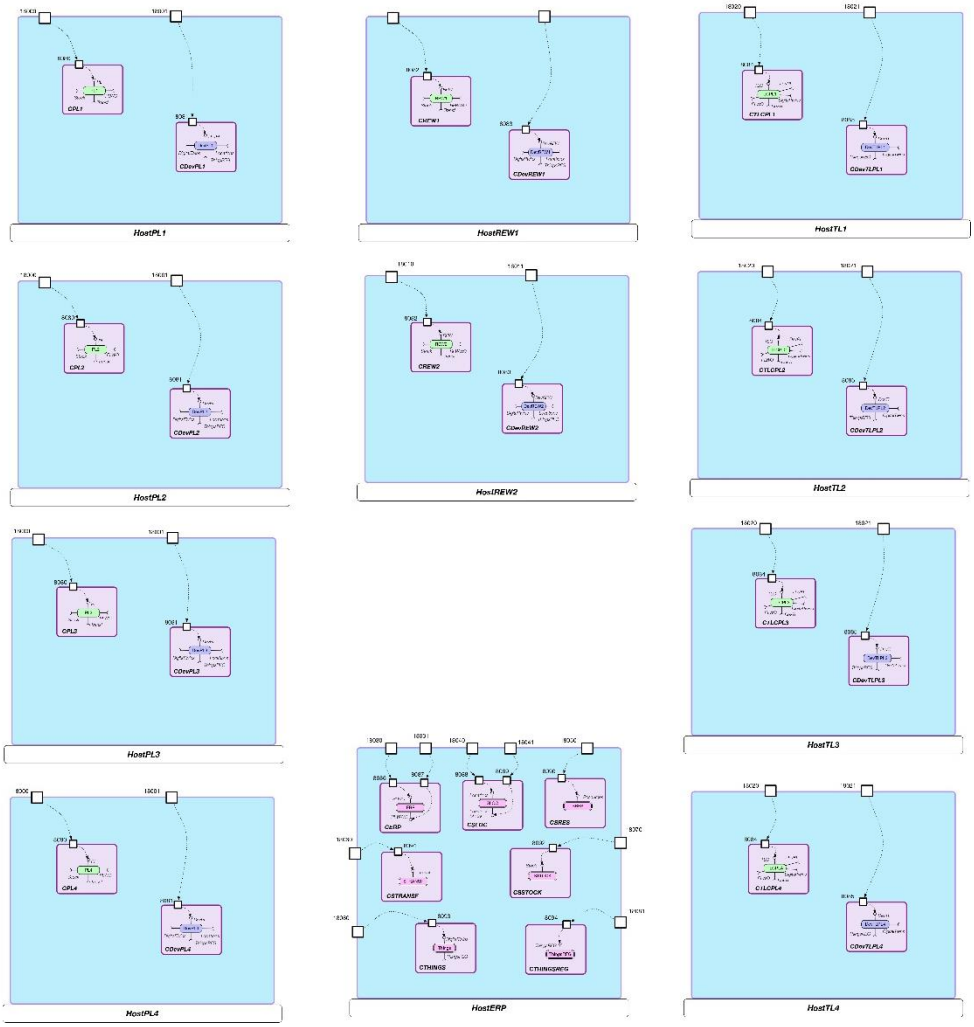

Figura 32: Distribució inicial de la solució industrial.

Com podem veure a l'esquema, cada *host* conté el servei que controla les accions del dispositiu, és a dir, el que realitza les funcions del dispositiu i el servei del dispositiu (degut a que aquests són tractats com a dispositius IoT i permeten interactuar amb el sistema) que registra l'estat del dispositiu i permet fer modificacions en la configuració d'aquest. A banda, com hem comentat al paràgraf anterior, els serveis generals els destinarem al "HostERP".

Per fer aquest desplegament inicial hem creat una regla d'adaptació anomenada "self-configure" que s'executa quan el monitor "self-configure" rep un "factory-reset". Aquest paràmetre provoca que la sonda "self-configure" envie una sol·licitud d'adaptació i ens genere el desplegament inicial. Per tant, si en un determinat moment necessitem reiniciar la simulació o tornar al punt de partida, sols necessitem que el monitor reba un "factory-reset" i s'iniciarà l'adaptació que farà les modificacions necessàries per tornar al punt de partida.

Arribats a aquest punt, tenim el que seria una primera aproximació de la solució, ara hem d'incloure els bucles de control a la solució que hem plantejat i adaptar els patrons i les funcionalitats a la solució. Comencem per veure com anem a incloure els bucles de control a la solució.

6.4. Bucles de control a l'escenari industrial

Per incloure els bucles de control, necessitem crear una sèrie de sondes monitors i regles d'adaptació que realitzen les modificacions necessàries en temps real.

Per a l'escenari industrial hem desenvolupat els següents elements:

- **Monitors**
  - **All Hosts Overloaded:** Monitoritza la càrrega computacional de tots els hosts per veure quan es troben tots sobrecarregats.
  - **Circuit Breaker:** Monitoritza les errades que poden cometre els µserveis.
  - **Load Balancer:** Monitoritza la càrrega computacional que estan sofrint els µserveis.
  - **Self Configuration:** Monitoritza la configuració inicial.
  - **Service Per Container:** Monitoritza l'estat del desplegament de la solució.

- **Sondes**
  - **All Hosts Overloaded:** Analitza i comprova que tots els *hosts* estiguen en un nivell correcte. Si en algun moment tots els *hosts* estan sobrecarregats, genera una petició per crear un host nou.
  - **Circuit Breaker:** Analitza i comprova que tots els µserveis funcionen correctament. Si en algun moment un µserveis comença a fallar, genera una petició per aplicar-li el patró *Circuit Breaker*.
  - **Load Balancer:** Analitza i comprova que cap µservei està sobrecarregat. Si en algun moment un µserveis comença a sobrecarregar-se, genera una petició per aplicar-li el patró *Load Balancer*.
  - **Self Configuration:** Analitza la solució i genera una petició per aplicar la configuració per defecte.

- **Service Per Container:** Analitza i comprova el desplegament de la solució, Si en algun moment es perd el patró *Service Per Container* genera una petició per tornar-lo a aplicat.

- **Regles**
  - **All Hosts Overloaded:** Conté les instruccions a executar per a crear un nou *host* en cas que els *hosts* disponibles es troben tots amb sobrecàrrega.
  - **Move Container:** Conté les instruccions per a moure un contenidor d'un *host* a un altre.
  - **Circuit Breaker:** Conté les instruccions per a aplicar un *Circuit Breaker* a un µservei determinat.
  - **Load Balancer:** Conté les instruccions per a aplicar un *Load Balancer* a un µservei determinat.
  - **Self Configuration:** Conté les instruccions per a aplicar la configuració per defecte a la solució.
  - **Service Per Container:** Conté les instruccions per a reorganitzar el desplegament de la solució amb el patró *Service Per Container*.

Aquests components estaran configurats de manera que es puguen utilitzar al escenari industrial. A més, les funcions haurem de tindre en compte que, com hi ha funcions que generen µserveis nous, hem de configurar les funcions per a que puguen generar els µserveis d'Embalpack com per exemple els PL o els DevPL. Per tant, adaptarem les regles per a que puguen determinar quin tipus de µservei ha de generar i generar-lo correctament per a que funcione a la solució industrial.

Una vegada configurat, hauríem de tindre tots els components necessaris per a que el sistema siga capaç d'utilitzar bucles de control. Sols ens quedarà distribuir els components al bucle MAPE-K de la mateixa manera que a la figura 22 i connectar la solució al bucle per a que puga ser gestionat adequadament.

Finalment ens quedà posar a prova el sistema per poder veure el funcionament de les funcions i els patrons que hem desenvolupat i configurat per a poder ser utilitzats a l'escenari industrial.

6.5. Escenaris per a la solució industrial

Hem desenvolupat una sèrie d'escenaris per veure com reacciona el sistema i comprovar que la solució i les funcionalitats funcionen correctament, però abans de començar a fer adaptacions comprovarem que el sistema s'inicialitza correctament.

El primer que farem serà arrancar la solució i veure que fa el sistema quan s'inicialitza. El primer que fa és iniciar els serveis del bucle de control. Una vegada iniciat el bucle de control, començarà a desplegar les regles, monitors, sondes, etc. que hem definit al nostre sistema. Finalment, desplegarà l'escenari inicial, que serà el mateix tant per als escenaris estàtics com per al dinàmic.

Per fer la implementació del bucle, FADA ens proporciona una manera d'alçar uns serveis, que ens permeten instal·lar totes les regles, monitors, etc. com si foren *plugins* OSGi de manera que ens permet fer la simulació dels escenaris com si es tractara d'un entorn real. Per tant, quan llancem l'execució de la solució, el primer que fa és desplegar el bucle de control i instal·lar aquests *plugins*, una vegada té la infraestructura preparada, la sonda llança un "factory reset" i s'inicia l'adaptació que desplega l'escenari inicial.

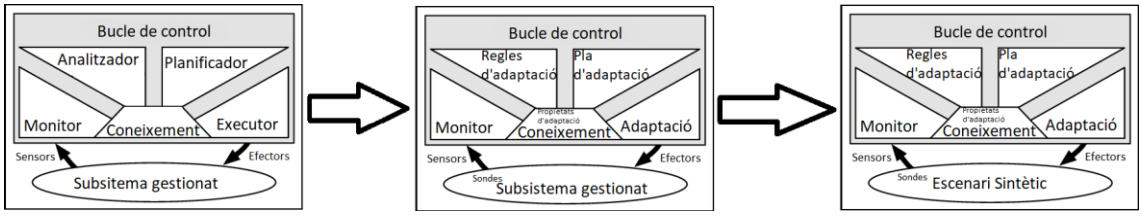

Figura 33: Seqüència de preparació de l'escenari inicial.

Com es pot apreciar a la figura anterior, primer es crearan els mòduls del bucle de control, després s'afegiran les regles, propietats d'adaptació, sondes, etc. que hem dissenyat per a la solució i, finalment, desplegarem l'escenari inicial.

6.5.1. Auto-Configuració inicial

A continuació, passarem a veure en detall com s'inicialitza el sistema de manera que podrem veure com es configuren i creen automàticament els diferents components del bucle, com es desplega el subsistema gestionat, i com s'inclouen els elements que permeten la interacció entre ambdós elements (monitors, propietats d'adaptació, regles d'adaptació, etc.).

En primer lloc, s'inicialitzarà el sistema, de manera que es desplegaran els diferents mòduls del bucle de control (Monitor, Analitzador, Planificador, Executor, Coneixement (línies 20 - 25)) i els components desenvolupats per a que aquest sistema puga treballar (monitors/sondes, propietats d'adaptació, regles d'adaptació i efectors línies 26 - 43)), per poder obtindre una solució on desplegar posteriorment l'escenari inicial.

Açò ens permet, una vegada s'inicien tots els components, fer ús d'un bucle de control funcional i adaptat al nostre cas d'estudi de manera que podem confirmar que, al afegir-li el cas d'estudi com a subsistema gestionat, els dos elements puguen treballar conjuntament de manera correcta.

```
20 202007022228290835 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [AnalyzingModule] BINDING SERVICE ...
21 202007022228290839 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [KnowledgegModule] GET SERVICE SUPPLY ...
22 202007022228290840 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [ExecutingModule] BINDING SERVICE ...
23 202007022228290840 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [ExecutingModule] DEPLOYING ...
24 202007022228290840 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [ExecutingModule] GET SERVICE SUPPLY ...
25 202007022228290841 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [PlanningModule] BINDING SERVICE ...
26 202007022228290844 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [MonitoringModule] GET SERVICE SUPPLY ...
27 202007022228290844 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [self-configure-monitor] BINDING SERVICE ...
28 202007022228290844 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [self-configure-monitor] DEPLOYING ...
29 202007022228290846 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [MonitoringModule] GET SERVICE SUPPLY ...
30 202007022228290847 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [circuitbreaker-monitor] BINDING SERVICE ...
31 202007022228290847 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [circuitbreaker-monitor] DEPLOYING ...
32 202007022228290849 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [MonitoringModule] GET SERVICE SUPPLY ...
33 202007022228290849 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [loadbalancer-monitor] BINDING SERVICE ...
34 202007022228290849 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [loadbalancer-monitor] DEPLOYING ...
35 202007022228290851 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [MonitoringModule] GET SERVICE SUPPLY ...
36 202007022228290851 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [host-load-level-monitor] BINDING SERVICE ...
37 202007022228290851 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [host-load-level-monitor] DEPLOYING ...
38 202007022228290860 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [AnalyzingModule] GET SERVICE SUPPLY ...
39 202007022228290860 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [SELF-CONFIGURE] BINDING SERVICE ...
40 202007022228290860 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [SELF-CONFIGURE] DEPLOYING ...
41 202007022228290862 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [AnalyzingModule] GET SERVICE SUPPLY ...
42 202007022228290863 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [circuitbreaker-configure-rule] BINDING SERVICE ...
43 202007022228290863 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [circuitbreaker-configure-rule] DEPLOYING ...
```

Figura 34: Desplegament del bucle de control.

Després desplegarem el subsistema gestionat on s'inicialitzarà el sistema amb la configuració predeterminada. La figura 34 mostra la part del *log* d'execució, on es veu com es llança aquesta inicialització (anomenada "Self Configure"), causada per un sensor que identifica que el sistema acaba d'arrancar i requereix una configuració "factory-reset" (línia 65). Açò desencadena una sèrie d'accions (segons el bucle de control MAPE-K) que fa que acabe llançant-se la regla "SelfConfigure" (línia 68), iniciant aquesta configuració.

Una vegada hem inicialitzat el procés de desplegament del subsistema que anem a gestionar, en aquest cas el sistema d'Embalpack, el sistema començarà a fer el desplegament dels diferents recursos que componen la solució de la manera més òptima possible que serà escollida pel sistema, i que està contemplat en un model segons la configuració proposada en la secció **¡Error! No se encuentra el origen de la referencia.** per la figura 32.

```
65  20200702222829091 9 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: factory-reset
66  20200702222829092 0 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: factory-reset
67  20200702222829097 7 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule SELF-CONFIGURE> Asking for the next System Configuration
68  20200702222830009 7 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule SELF-CONFIGURE> Request Planning (c6835499c3a949879a63041429698788)
69  20200702222830010 2 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (c6835499c3a949879a63041429698788) <<<
70  20200702222830037 7 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DEPLOYING HOST
71  20200702222830038 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] DEPLOYING HOST
72  20200702222830038 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL3] DEPLOYING HOST
73  20200702222830038 5 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL4] DEPLOYING HOST
74  20200702222830038 6 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW1] DEPLOYING HOST
75  20200702222830038 7 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW2] DEPLOYING HOST
76  20200702222830038 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL1] DEPLOYING HOST
77  20200702222830038 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] DEPLOYING HOST
78  20200702222830038 9 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DEPLOYING HOST
79  20200702222830039 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL4] DEPLOYING HOST
80  20200702222830039 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DEPLOYING HOST
81  20200702222830039 2 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] EXPOSING EDNPOINT 18000
82  20200702222830039 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] EXPOSING EDNPOINT 18001
83  20200702222830039 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] EXPOSING EDNPOINT 18000
84  20200702222830039 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] EXPOSING EDNPOINT 18001
85  20200702222830039 4 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL3] EXPOSING EDNPOINT 18000
86  20200702222830039 4 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL3] EXPOSING EDNPOINT 18001
87  20200702222830039 4 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL4] EXPOSING EDNPOINT 18000
88  20200702222830039 5 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL4] EXPOSING EDNPOINT 18001
89  20200702222830039 5 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW1] EXPOSING EDNPOINT 18010
90  20200702222830039 5 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW1] EXPOSING EDNPOINT 18011
91  20200702222830039 5 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW2] EXPOSING EDNPOINT 18010
92  20200702222830039 6 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW2] EXPOSING EDNPOINT 18011
93  20200702222830039 7 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL1] EXPOSING EDNPOINT 18021
```

Figura 35: Desplegament del subsistema gestionat.

En aquest cas, segons l'extracte que es mostrarà a continuació, primer es desplegaran els *hosts* (línies 70 - 80) i a continuació els *enpoints* requerits (línies 81 - 93).

Finalment, el sistema acabarà el desplegament i es quedarà a l'espera, monitoritzant la situació global del subsistema gestionat, de manera que en cas que es produïsca una anomalia, aquesta siga detectada per les sondes i es procedisca a actuar d'una determinada manera per poder solucionar aquesta anomalia i tornar a un estat òptim del sistema.

Una vegada acabat el desplegament del bucle de control i el subsistema gestionat, obtindrem una solució inicial com la de la següent figura, que correspon a la seua vegada amb l'especificació de la figura 32.

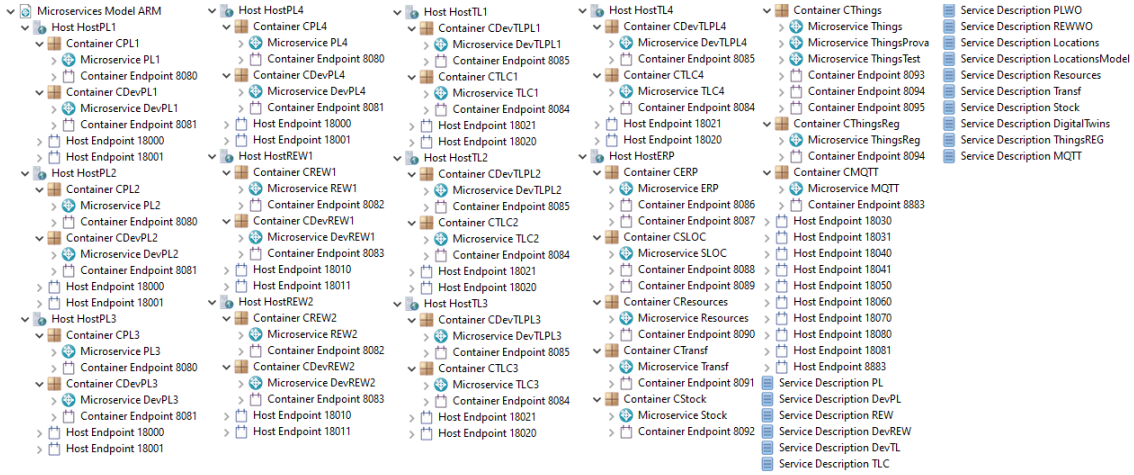

Figura 36: Escenari inicial.

A continuació es mostraran una sèrie d'escenaris on es presenten algunes de les situacions o anomalies per a les que hem preparat mesures i mètodes que ens permeten solucionar-les. Per poder contemplar tots els casos, hem conduït al sistema a algunes situacions. A més, es mostra un cas on el sistema evoluciona gràcies a un procés de simulació aleatòria, per avaluar com s'autogestiona i, per tant, les situacions observades no estan controlades amb anterioritat, pel que no podem saber d'avantmà com evolucionarà i s'autoadaptarà el sistema pera gestionar aquestes situacions.

6.5.2. Aplicació del Load Balancer i del Circuit Breaker

El primer escenari que hem descrit ens comenta la següent situació:

"Ens trobem a l'inici de la jornada, iniciem el sistema i inicialitzem la configuració inicial. En un determinat moment, ens arriba una petició de producció elevada i la cadena de producció nº1 comença a augmentar la càrrega de treball exponencialment. Quan aquesta aconsegueix un nivell de càrrega elevat, la sonda encarregada de monitoritzar els µserveis implicats a la cadena nº1 detecta que un µserveis té molta càrrega de treball i decideix aplicar-li un balancejador de càrrega de manera que llança la regla d'adaptació que realitza aquest canvi i així aconsegueix reduir el nivell d'estrés del µservei.

Després d'una estona, un µservei a la cadena nº2 comença a donar una sèrie d'errors relacionats amb la connexió i, per tant, els µserveis que depenen d'aquest comencen a obtindre respostes errònies. La sonda capta aquesta anomalia i decideix aplicar un *Circuit Breaker* de manera que, mitjançant aquesta tècnica de prevenció, aconsegueix mitigar l'impacte de la pèrdua de funcions i així simular que el sistema continua funcionant correctament fins que aquest µservei torne a funcionar correctament."

Per tant, seguint el plantejament de l'enunciat anterior, el primer que farà el sistema és aplicar un patró *Load Balancer* al µserveis PL1, per tant realitzarà les accions comentades a capítols anteriors, on construíem un contenidor per al servei que farà la funció de balancejador de càrrega. També farem 3 còpies del µservei ja que hem detectat un percentatge de càrrega elevat, depenent del percentatge, es crearan més còpies o menys.

Després d'aplicar la regla del *Load Balancer*, se'ns indica que és necessari aplicar un *Circuit Breaker* al µservei PL2. Per tant, llançarem el procés d'adaptació per aplicar aquest patró, on seguint l'estratègia de desplegament *Service per Container* i realitzant les accions comentades a capítols anteriors on comentàvem com es desplega un patrò *Circuit Breaker*, realitzarem les accions necessàries per a fer que l'adaptació es realitze correctament.

En aquesta ocasió, els patrons s'han aplicat al mateix *host* que estaven allotjats originalment els µserveis, per tant, el resultat a obtindre, després d'aplicar les dos adaptacions comentades anteriorment, serà el que es mostrarà a la següent imatge:

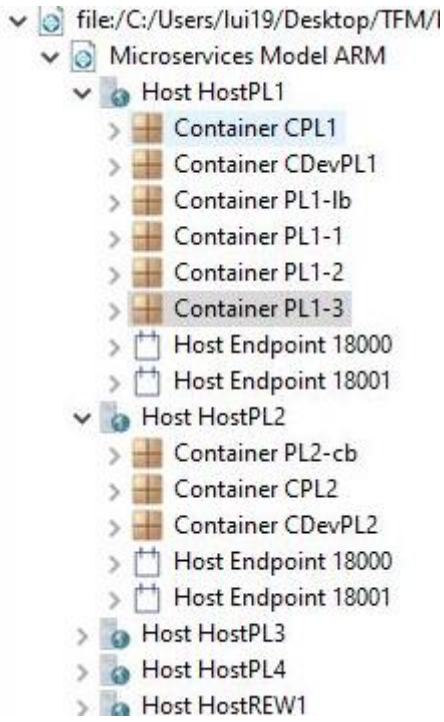

Figura 37: Escenari *Load Balancer* i *Circuit Breaker*.

A continuació passarem a veure en detall el procés que acabem de comentar:

Després d'inicialitzar el sistema i obtindre escenari com el que hem vist abans, el sistema rep una anomalia i decideix iniciar una adaptació per a resoldre'l i mantenir així el sistema en un estat òptim. En aquest cas, hem forçat, en primer lloc, una adaptació de tipus *Load Balancer* per poder veure si aquest mecanisme funciona correctament.

```
470  20200702228300791 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
471  20200702228300960 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (c6835499c3a949879a63041429698788) <<<
472
473  osgi> lb PL1 high
474
475  20200702230410676 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: PL1 => high
476  20200702230410679 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: PL1 => high
477  20200702230410681 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule LOAD-BALANCER> Asking for the next System Configuration
478  20200702230410694 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule LOAD-BALANCER> Request Planning (2de0aafe1f4148aab1947a12ea5e97b0)
479  20200702230410695 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (2de0aafe1f4148aab1947a12ea5e97b0) <<<
480  osgi> 20200702230410761 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DEPLOYING CONTAINER PL1-lb
481  20200702230410765 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-lb] DEPLOYING CONTAINER PL1-lb IN HOST HostPL1
482  20200702230410766 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW1] DEPLOYING CONTAINER PL1-1
483  20200702230410767 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-1] DEPLOYING CONTAINER PL1-1 IN HOST HostREW1
484  20200702230410767 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL4] DEPLOYING CONTAINER PL1-2
485  20200702230410768 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-2] DEPLOYING CONTAINER PL1-2 IN HOST HostPL4
486  20200702230410769 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL4] DEPLOYING CONTAINER PL1-3
487  20200702230410769 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-3] DEPLOYING CONTAINER PL1-3 IN HOST HostTL4
488  20200702230410770 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DELEGATE ENDPOINT 18000 TO CONTAINER PL1-lb ENDPOINT 8080
```

Figura 38: Adaptació *Load Balancer* en PL1.

Com es pot veure a la imatge anterior, una vegada el monitor rep un valor anòmal, en aquest cas "high", inicia el procés d'adaptació per a resoldre la situació. En aquest cas, desplegarà el contenidor per a ubicar el *Load Balancer* (línia 480) i, a més, farà una sèrie de còpies del µservei afectat, en aquest cas, PL1 (línies 482, 484 i 486).

Finalment, acabarà el procés d'adaptació i aplicarà els canvis a la configuració (línia 528), açò es fa així ja que, com hem comentat abans, sempre realitzem les adaptacions a una còpia del model per assegurar l'estabilitat de la solució i una vegada comprovem que l'adaptació és correcta, actualitzem el model sobre el que està treballant el sistema.

```
524  202007022230410781 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-2] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
525  202007022230410781 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-3] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
526  202007022230410781 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-3] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
527  202007022230410781 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-3] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
528  202007022230410782 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
529  202007022230410867 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (2de0aafe1f4148aab1947a12ea5e97b0) <<<
```

Figura 39: Adaptació completada.

Finalment, una vegada s'ha actualitzar el model sobre el que treballa el sistema i hem arribat a un estat on l'adaptació ha sigut exitosa i el sistema ha sigut actualitzat correctament aconseguint així un estat òptim, el sistema aplicarà qualsevol adaptació que es trobe a l'espera o en cas de no haver-ne cap, esperarà a aquest nou estat fins que es requerisca d'una nova adaptació.

En aquest cas el sistema sí requereix d'una nova adaptació ja que, com bé ens indica el enunciat, necessitem aplicar una adaptació de tipus *Circuit Breaker*. A continuació veurem l'extracte on es realitza aquesta nova adaptació:

```
531  osgi> cb PL2 high
532
533  202007022231250767 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostPL2)[Container: CPL2]<Microservice: PL2> => high
534  202007022231250768 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostPL2)[Container: CPL2]<Microservice: PL2> => high
535  202007022231250769 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule circuitbreaker-configure-rule> Asking for the next System Configuration
536  202007022231250775 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule circuitbreaker-configure-rule> Request Planning (f918ff0fb7f1426e99cbce10fd2fef11)
537  202007022231250776 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (f918ff0fb7f1426e99cbce10fd2fef11) <<<
538  osgi> 202007022231250835 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DEPLOYING CONTAINER PL2-cb
539  202007022231250835 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL2-cb] DEPLOYING CONTAINER PL2-cb IN HOST HostPL1
540  202007022231250836 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DELEGATE ENDPOINT 18000 TO CONTAINER PL2-cb ENDPOINT 8080
541  202007022231250836 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL2-cb] EXPOSING ENDPOINT 8080
542  202007022231250836 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL2-cb] DEPLOYING MICROSERVICE PL2-cb
543  202007022231250837 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-cb] DEPLOYING MICROSERVICE PL2-cb IN CONTAINER PL2-cb
544  202007022231250837 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-cb] EXPOSING SERVICE PL THROUGH INTERFACE pl
545  202007022231250837 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-cb] DEFINING REQUIRED SERVICE PL THROUGH INTERFACE reqpl
546  202007022231250837 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL2-cb] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL2-cb ENDPOINT pl
547  202007022231250838 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-cb] PUBLIC BINDING reqpl TO HOST HostPL2 ENDPOINT 18000
548  202007022231250838 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
549  202007022231250897 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (f918ff0fb7f1426e99cbce10fd2fef11) <<<
```

Figura 40: Adaptació Circuit Breaker en PL2.

Com es pot observar, el procés d'adaptació inicia igual que el cas anterior, és a dir, s'observa una anomalia al sistema i el bucle de control decideix iniciar un procés d'adaptació. En aquest cas al ser un procés més curt, es poden observar totes les accions que realitza el sistema per a aplicar l'adaptació.

En primer lloc, el sistema desplega un contenidor per ubicar el nou µservei que farà l'acció del *Circuit Breaker* (línia 538). Seguidament crearà els recursos necessaris per a que aquest sistema estiga visible i es puga utilitzar (línies 539 - 546). Després crearà els *bindings* necessaris per a que es puga comunicar amb el recurs afectat (línia 547).

Finalment, ens indicarà que l'adaptació és correcta i finalitzarà el procés d'adaptació quedant-se a l'espera de futures adaptacions.

6.5.3. Aplicació del moviment de recursos

Al segon escenari observarem que aquest conté unes circumstàncies similars a les del primer escenari però amb alguna novetat addicional. Aquest ens comenta la següent situació:

"Ens trobem a l'inici de la jornada, iniciem el sistema i inicialitzem la configuració inicial. En un determinat moment, la cadena de producció nº2 comença a realitzar feines de comprovació de funcionalitats i manteniment mentre continua amb els treballs que té assignats. Aquest protocol provoca que el *host* de la cadena nº2 comence a saturar-se i s'observe un augment de la càrrega al *host*. La sonda que està monitoritzant el sistema observa que hi ha disponibles diversos *hosts* on la càrrega computacional és baixa i decideix llançar una regla d'adaptació per moure alguns µserveis a algun d'aquests *hosts* menys carregats i així alleugerar la càrrega computacional del *host* de la cadena nº2.

Després d'una estona, un µservei a la cadena nº1 comença a donar problemes i degut a açò, el funcionament de la cadena de producció comença a veure's afectat. La sonda s'adona que aquest µservei a superat el llindar d'errades permeses i decideix llançar una regla d'adaptació per aplicar el patró *Circuit Breaker* fins que aquest µservei torne a la normalitat."

Com podem observar, és un cas similar al del primer escenari però en aquest cas afegim l'adaptació on es mouen alguns contenidors a un altre *host* per alliberar càrrega computacional.

Per a analitzar aquest escenari utilitzarem el model que genera al final de les adaptacions esmentades al paràgraf anterior ja que el monitor ens tornarà un informe paregut al de l'escenari 1, però amb la informació de les adaptacions de l'escenari 2, per tant aprofitarem que podem visualitzar el model per veure d'una manera més senzilla l'adaptació que ha realitzat.

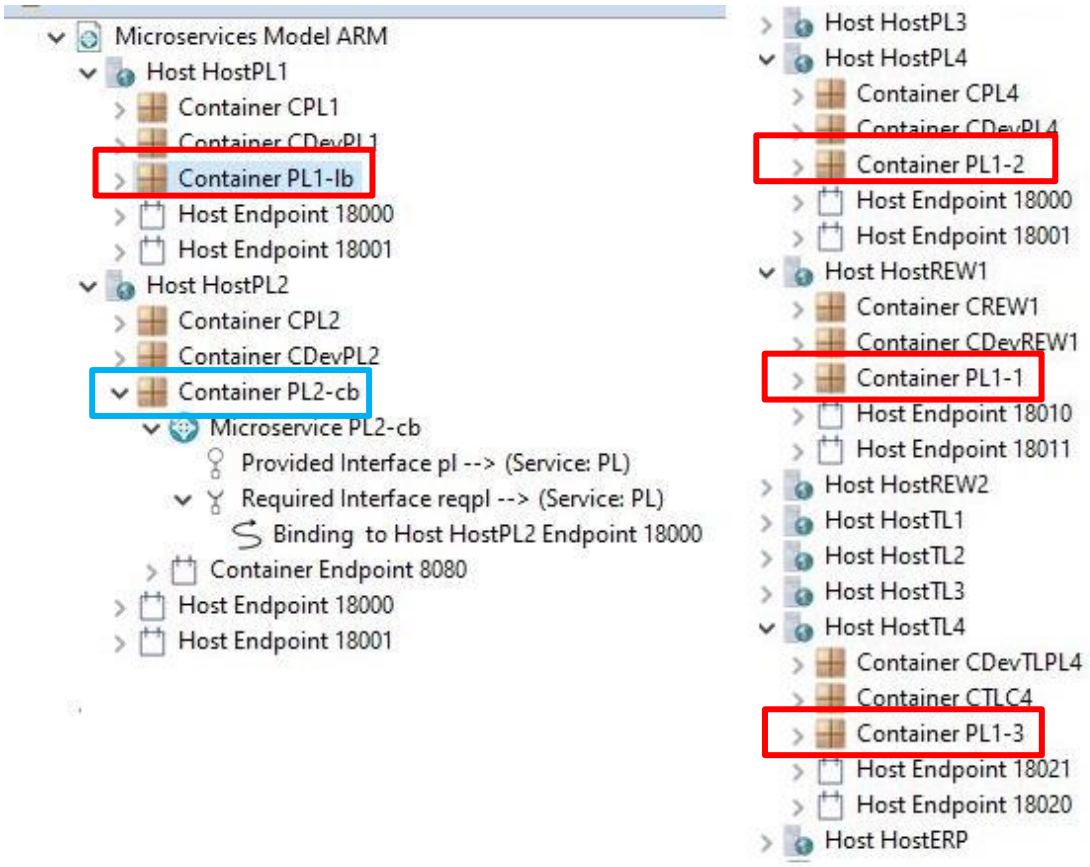

Figura 41: Escenari *moviment de recursos*.

Com es pot veure a la figura 41, el primer que ha fet el sistema és generar el *Load Balancer* i distribuir les còpies als *hosts* que presenten menys càrrega computacional (ressaltat en roig). Seguidament ha aplicat un *Circuit Breaker* al µservei "PL2" que és el que presentava els errors detectats.

A continuació veurem com ha decidit el sistema fer el moviment de recursos ja que pot o bé moure el recurs una vegada desplegat o moure'l al nou *host* i després desplegar-lo:

```
492  20200702225751012l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-moved] EXPOSING SERVICE PL THROUGH INTERFACE pl
493  20200702225751012l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-moved] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
494  20200702225751012l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-moved] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
495  20200702225751012l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-moved] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
496  20200702225751012l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL2-moved] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL2-moved ENDPOINT pl
497  20200702225751012l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-moved] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
498  20200702225751012l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-moved] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
499  20200702225751013l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-moved] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
```

Figura 42: Moviment de recursos al Load Balancer.

Com es pot apreciar a la figura 42, en aquesta ocasió el sistema ha decidit que per millorar l'eficiència de les adaptacions el millor era moure els recursos als *hosts* menys carregats i desplegar allí els diferents recursos. Per poder diferenciar els elements que han sigut moguts, el que hem fet ha sigut afegir-li una etiqueta "moved" de manera que els que són etiquetats amb aquesta etiqueta corresponen als recursos que han sigut moguts de *host* per optimitzar la càrrega computacional.

6.5.4. Aplicació del Service Per Container

Finalment l'últim escenari és d'adaptació a un mètode de desplegament. S'ha observat que degut a les actualitzacions que han realitzat els tècnics, s'ha perdut el patró de desplegament per tant s'ha decidit llançar la regla d'adaptació que redistribueix la solució i aplica el patró *Service per Container*.

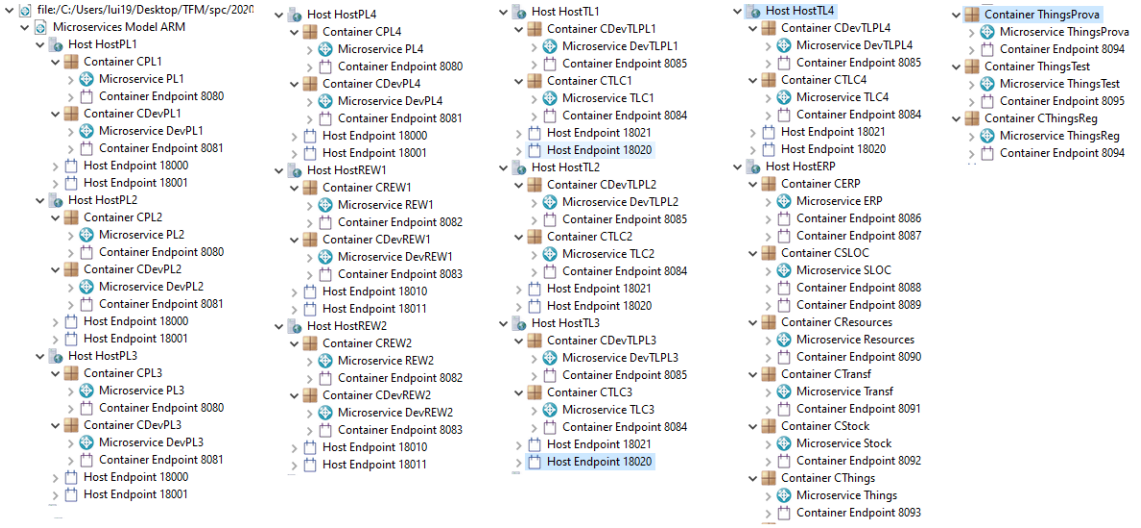

Figura 43: Escenari *Service Per Container*.

Com es pot veure a la figura 43, si ens fixem, tots els *hosts* tenen contenidors a dintre seu i a dintre dels contenidors sols hi ha un µservei, per tant podem afirmar que el patró que hem aplicat amb l'adaptació del sistema s'ha aplicat correctament.

A continuació passarem a veure amb més detall com s'aplica el patró *Service per Container*:

```
485  20200708172730072l [INFO ] ....lite.artifacts.components.Probe: (service-per-container-probe) Reporting measure: service-per-container => alert
486  20200708172730072l [DEBUG] ...ite.artifacts.components.Monitor: (Service per Container Monitor) Received measure: service-per-container => alert
487  20200708172730072l [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule service-per-container-rule> Asking for the next System Configuration
488  20200708172730073l [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule service-per-container-rule> Request Planning (68ad15c79e284d0ca14c277cc6c4d0dd)
489  20200708172730073l [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (68ad15c79e284d0ca14c277cc6c4d0dd) <<<
490  20200708172730078l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] REMOVE ENDPOINT DELEGATION 8095
491  20200708172730078l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] REMOVE ENDPOINT DELEGATION 8094
492  20200708172730078l [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DEPLOYING CONTAINER ThingsTest-moved
493  20200708172730078l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [ThingsTest-moved] DEPLOYING CONTAINER ThingsTest-moved IN HOST HostERP
494  20200708172730078l [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DEPLOYING CONTAINER ThingsProva-moved
495  20200708172730078l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [ThingsProva-moved] DEPLOYING CONTAINER ThingsProva-moved IN HOST HostERP
496  20200708172730078l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] REMOVING ENDPOINT 8095
497  20200708172730078l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] REMOVING ENDPOINT 8094
498  20200708172730078l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [ThingsTest-moved] EXPOSING ENDPOINT 8095
499  20200708172730078l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [ThingsProva-moved] EXPOSING ENDPOINT 8094
500  20200708172730078l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] UNDEPLOYING MICROSERVICE ThingsTest
501  20200708172730078l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [ThingsTest] UNDEPLOYING MICROSERVICE
502  20200708172730079l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] UNDEPLOYING MICROSERVICE ThingsProva
```

Figura 44: Adaptació *Service per Container*.

Com es pot veure a la figura 44, el que fa el sistema es moure els µserveis que no estan ubicats correctament i per tant, com es una operació de moviment, etiquetarà els µserveis moguts amb "moved". A més com podem veure a la imatge, el sistema ha detectar que sols els components que es mostren estan mal ubicats i per tant l'adaptació sols s'aplica a aquests obviant la resta de la solució que sí es troba ubicada correctament.

6.5.5. Aplicació d'adaptacions autònomes

Aquest escenari mostra una execució on el sistema s'inicialitza amb valors de càrrega aleatoris, i es deixa el sistema durant un temps que se li apliquen totes les regles d'adaptació que li puguen esdevindre, segons s'han definit al treball. El que farem, per tant, és executar el prototip i en compte de comprovar que el resultat que hem forçat és el que hem obtingut, analitzarem els canvis que s'han produït i comprovarem que els canvis són correctes.

Tal vegada aplique diversos patrons o realitze una redistribució del sistema diverses vegades però, com és dinàmic, cada vegada que executem ens crea un escenari diferent i, per tant, és pràcticament impossible saber com reaccionarà el sistema, degut a que la generació de les característiques inicials del escenari són aleatòries. A continuació, mostrarem el resultat de l'execució on s'han realitzat més de quinze adaptacions.

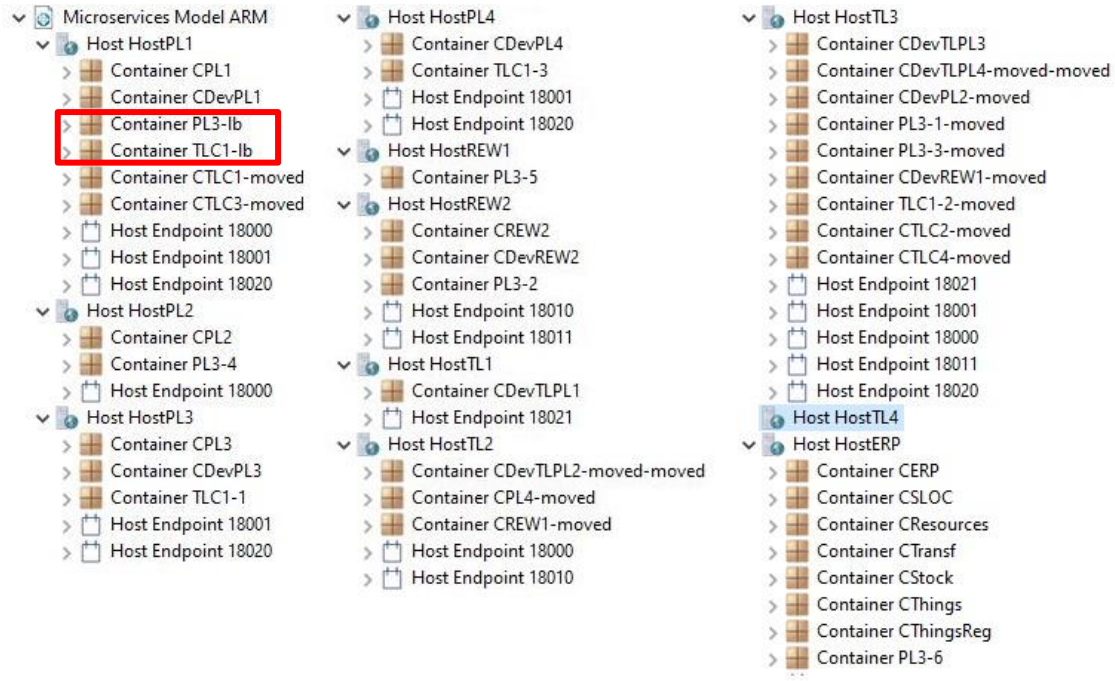

Figura 45: Escenari autònom.

Com es pot comprovar a la figura 45, després de realitzar diverses adaptacions podem observar que el sistema ha aplicat diversos *Load Balancer* (ressaltats en roig) i a més ha estat movent diversos µserveis depenent de la càrrega computacional que els *hosts* tingueren en un determinat moment. També podem observar que al no ser un escenari executat en un entorn real, sinó que ha sigut una simulació, no tenim manera de simular que un µservei falle, per tant, el bucle de control no ha aplicat cap *Circuit breaker*.

A continuació, veurem què ha passat al sistema per a acabar amb la configuració de la figura 38 i quines són les peculiaritats dels sistemes autònoms.

```
475  20200702230317091 9 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: PL3 => very-high
476  20200702230317092 1 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: PL3 => very-high
477  20200702230317092 3 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule LOAD-BALANCER> Asking for the next System Configuration
478  20200702230317093 7 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule LOAD-BALANCER> Request Planning (660fe783b71940a2b0ad450a4880847a)
479  20200702230317094 0 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: DevTLPL1 => low
480  20200702230317094 1 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (660fe783b71940a2b0ad450a4880847a) <<<
481  20200702230317094 1 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: DevTLPL1 => low
482  20200702230317094 2 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: TLC1 => high
483  20200702230317094 2 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: TLC1 => high
484  20200702230317094 3 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule LOAD-BALANCER> Asking for the next System Configuration
485  20200702230317095 2 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule LOAD-BALANCER> Request Planning (acdaf853228640798a29df02adb9274b)
486  20200702230317095 4 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: TLC3 => low
487  20200702230317095 4 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: TLC3 => low
488  20200702230317095 5 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: things => low
489  20200702230317095 5 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: things => low
490  20200702230317095 8 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostTL1) => 2.2666667
491  20200702230317095 9 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostTL1) => 2.2666667
492  20200702230317096 1 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
493  20200702230317097 0 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (f0e941f7cc244499a5eee08f0cdf4abf)
494  20200702230317097 1 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostTL2) => 1.6666666
495  20200702230317097 1 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostTL2) => 1.6666666
496  20200702230317097 5 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
497  20200702230317098 2 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (a8c8c472550b4ec1b92e0963d84adfae)
498  20200702230317098 2 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostTL3) => 1.117647
499  20200702230317098 3 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostTL3) => 1.117647
500  20200702230317098 4 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
501  20200702230317099 5 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (e6cba86c2e6f497791918bd799f24b3d)
```

Figura 46: Adaptació autònoma.

Com podem apreciar a la figura 46, a un sistema autònom poden ocórrer més d'una adaptació abans que aquesta finalitze. Açò provoca que mentre una adaptació esta produint-se o s'està iniciant, apareguen noves anomalies i s'emmagatzemen a la base de dades per ser resoltes.

Degut a aquesta peculiaritat del sistema hem de tindre en compte que hem d'executar les peticions d'adaptació en l'ordre que es demanaren, per tant, el que fem és encuar les peticions a una cua de manera que una vegada resolta l'adaptació, es comprove la cua i si hi ha alguna petició es realitze fins que es buide la cua i ens quedem a l'espera de noves adaptacions.

Com es pot observar a la imatge anterior, cada petició d'adaptació compta amb un identificador únic, de manera que afegint aquest identificador a una cua d'adaptacions podem fer una gestió adequada de les adaptacions pendents. Per tant, l'únic que necessitem és consultar aquesta cua i realitzar les peticions que hi apareguen d'una manera ordenada.

6.6. Resultats

Després de veure els resultats que presenten els escenaris anteriors passarem a analitzar els resultats i veure si hem aconseguit un resultat satisfactori.

Prèviament ja hem pogut comprovar que conceptualment les tècniques detallades anteriorment són aplicables i per tant la arquitectura de µserveis és capaç de treballar junt amb computació autònoma. Açò ho hem demostrat de manera teòrica però, a més, hem decidit utilitzar un entorn industrial per demostrar que aquestes tècniques es poden utilitzar més enllà d'un entorn teòric.

Com podem veure als quatre escenaris detallats al capítol anterior, el resultat de l'execució del sistema és satisfactori. Hem aconseguit que els escenaris es comportaren com estava previst i hem obtingut un sistema amb capacitat autònoma capaç de realitzar adaptacions en calent i modificar la seua estructura per adaptar-se a les circumstàncies.

Com podem veure als dos primers escenaris hem aconseguit que s'aplicaren els patrons *Load Balancer* i *Circuit Breaker* correctament i a més, al segon escenari, el sistema ha redistribuït els µserveis per a disminuir la càrrega computacional dels *hosts*. Al tercer escenari, observem que la solució s'havia desplegat malament i, per tant, s'ha realitzat una adaptació per a tornar a desplegar el sistema.

Finalment, la solució dinàmica ha anat realitzant les adaptacions de forma autònoma adaptant-se a les necessitats del sistema. Podem veure que els quatre escenaris han fet el que s'esperava i ho han fet de manera autònoma i correcta així que podem afirmar que la tècnica descrita conceptualment és transferible i utilitzable en un entorn industrial.

7. Conclusions

Una vegada finalitzat el projecte podem donar un pas enrere i veure que hem obtingut i quines conclusions podem obtindre després d'haver fet totes les proves i haver analitzat els resultats.

Com aquest experiment l'hem dissenyat en dues parts per demostrar que les tècniques mostrades són aplicables a qualsevol entorn i després veure els efectes que produiria en un entorn industrial, dividirem les conclusions en dues parts també.

En primer lloc, després de fer un treball de recerca per veure les característiques, funcionalitats, avantatges i inconvenients, impacte, etc. de la computació autònoma i de l'arquitectura basada en µserveis, hem aconseguit obtindre diverses maneres d'unir aquestes dos tècniques de desenvolupament de programari. Durant aquest projecte hem vist diversos patrons de disseny de µserveis on hem sigut capaços d'adaptar alguns d'aquests per a que treballen amb computació autònoma. Finalment, després de dissenyar l'experiment i analitzar els resultats de la recerca, l'anàlisi i el disseny podem afirmar que conceptualment, aquestes dos tècniques es poden utilitzar conjuntament i podrien tindre un impacte beneficiós en la indústria.

En segon lloc, després de realitzar diverses proves sobre l'escenari industrial podem concloure que el sistema ha funcionat com s'esperava i és capaç d'adaptar-se segons les necessitats i respondre davant de les incidències que és capaç de detectar. Hem realitzat proves forçant el sistema per a produir tres escenaris preestablerts i el sistema ha aconseguit acabar amb una solució adaptada a les circumstàncies però amb totes les funcionalitats que teníem a l'inici. A més, hem deixat que el sistema realitzara adaptacions automàticament i també ha donat un resultat satisfactori, mantenint les funcionalitats i adaptant-se a les circumstàncies. Hem deixat el sistema funcionant durant una estona i després de forçar incidències al sistema, aquest ha aconseguit reparar-les, per tant pensem que aquest ús de les tècniques de desenvolupament de programari i computació autònoma podrien ser beneficiós en un entorn industrial.

Com a conclusió d'aquest projecte, ens podem donar per satisfets ja que les proves realitzades han donat un resultat adequat i ens han permés demostrar la hipòtesi establerta a l'inici del projecte. Per descomptat, nosaltres hem desenvolupat una xicoteta part de les funcionalitats, sempre es podria continuar amb el desenvolupament i augmentar les capacitats d'adaptació del sistema.

7.1. Conclusions personals

Com a conclusió personal, sols em queda afegir les impressions personals i que he extret després d'haver fet el desenvolupament i la recerca.

Després d'haver-me enfrontat a aquest repte, puc afirmar que aquestes dos tècniques poden resultar beneficioses i podrien ser d'utilitat en un futur. A més, després de fer la recerca i el desenvolupament posterior, he aconseguit aprendre diverses tècniques de desenvolupament de programari i he descobert una aplicació dels models i dels µserveis que no coneixia i pense que hem pot resultar útil en un futur pròxim.

Considere que, gràcies a la realització d'aquest projecte, he adquirit experiència en el camp de la producció de mètodes de programari i he descobert tècniques que són aplicables a qualsevol entorn.

Finalment, cal afegir que, gràcies a l'ajuda del tutor, he aconseguit que la fase d'aprenentatge de les tècniques utilitzades i de recerca d'informació haja resultat més lleugera del que pensava en un principi, el que m'ha permès obtindre un enfocament diferent i afrontar els reptes que m'ha suposat el projecte d'una manera més assequible i amena.

7.2. Relació amb els estudis

Aquest projecte està relacionat directament amb la assignatura "intel·ligència ambiental" ja que és l'assignatura on se'ns van presentar els sistemes IoT, els µserveis i els protocols de comunicació distribuïda. A més, a "Serveis i aplicacions distribuïdes" se'ns introdueix al món dels sistemes distribuïts i ens ensenya com es comuniquen aquests sistemes entre altres coses.

També podem trobar una relació amb l'assignatura "Sistemes informàtics per al control d'instal·lacions i processos" ja que se'ns mostren diversos tipus de comunicacions industrials i distribuïdes.

7.3. Treballs futurs

Per fer diverses ampliacions, el que faríem seria continuar amb l'estudi de funcionalitats, patrons i tècniques de l'Arquitectura de µserveis per veure quines podríem aplicar al camp de la computació autònoma i així augmentar les possibilitats del sistema.

Una altra opció seria anar més enllà i estudiar quins altres camps, a banda de l'Arquitectura de µserveis, serien viables per utilitzar conjuntament amb aquestes dos tècniques i així augmentar més l'escalabilitat del sistema.

Per fer tot açò s'haurien de fer proves de la mateixa manera que les hem fetes en aquest projecte i determinar si tindrien un impacte positiu i els beneficis que s'obtindrien amb la implantació d'aquest tipus de sistemes.

8. Bibliografia

[1] IBM. *Service Oriented Architecture*. IBM Knowledge Centre, Technical Report, 2014. [En línia]. https://www.ibm.com/support/knowledgecenter/es/SS8JB4/com.ibm.wbpm.wid.main.doc/prodoverview/topics/csoa.html. [Últim accés: 09/2020]

[2] S.J. Fowler, *Production Ready Microservices*. Ed. O-Really, ISBN 978-1-491-96597-9, 2017

[3] M. Fowler. *Microservices: a Definiton of this New Architectural Term*. 2014. [En línia]. https://martinfowler.com/articles/microservices.html. [Últim accés: 2020]

[4] MuleSoft, *Best Practices for Microservices (White Paper)*, Technical report, 2020. [En línia]. https://www.mulesoft.com/lp/whitepaper/api/microservices-best-practices

[5] C. Richardson. *Microservices Patterns*. Ed. Manning, 520 pags, ISBN 9781617294549. Octubre, 2018. [En línia]. https://www.manning.com/books/microservices-patterns. [Últim accés: 09/2020]

[6] C. Richardson. *Microservices: From Design to Deployment*. NGNIX, eBook, 2016.

[7] Microsoft. *Estilo de arquitectura de microservicios*. 2019. [En línia]. https://docs.microsoft.com/es-es/azure/architecture/guide/architecture-styles/microservices. [Últim accés: 09/2020]

[8] C. Stetson, *Microservices: Reference Architecture*, NGNIX, eBook, 2017

[9] Y. Brun et. al. *Engineering Self-Adaptive Systems through Feedback Loops Roadmap.* Self-Adaptive Systems, Ed. Springer-Verlag Berlin Heidelberg, LNCS 5525, pp. 48-70, 2009

[10] R. de Lemos et. al. *Software Engineering for Self-Adaptive Systems: A Second Research Roadmap*. Self-Adaptive Systems, Ed. Springer-Verlag Berlin Heidelberg, LNCS 7475, pp. 1–32, 2013

[11] IBM, *An architectural blueprint for Autonomic Computing (4th Edition)*, June 2006

[12] T.B. Sheridan. *On how often the Supervisor should sample*. IEEE Transactionson Systems Science and Cybernetics, Vol 6(2), pp. 140–145, 1970

[13] B. Chen et. al, *Software Engineering for Self-Adaptive Systems: A Research Roadmap*. Self-Adaptive Systems, Ed. Springer-Verlag Berlin Heidelberg, LNCS 5525, pp. 1-26, 2009

[14] M. Salehie, L. Tahvildari. *Self-Adaptive Software: Landscape and Research Challenges*. ACM Transactions on Autonomous and Adaptive Systems, Vol. 4 (2), Article 14, May 2009

[15] J. Andersson, R. de Lemos, S. Malek, D. Weyns. *Modeling Dimensions of Self-Adaptive Software Systems*. Self-Adaptive Systems. LNCS 5525, pp. 27–47. Springer Verlag Heidelberg, 2009

[16] J. Kramer, J. Magee. *Self-Managed Systems: an Architectural Challenge*. Future of Software Engineering, pp. 259–268. IEEE Computer Society, Washington, USA, 2007

[17] S. Dobson, et al. *A Survey of Autonomic Communications*. ACM Transactions on Autonomous and Adaptive Systems (TAAS) Vol 1, pp. 223–259, 2006

[18] O. Selma, S. Boulehouache, S. Mazouzi. *A Survey of Uncertainties in MAPE-K Control Loop*. International Conference on Advanced Technologies, Computer Engineering and Science (ICATCES'18), May 11-13, 2018

[19] Y. Abuseta and K. Swesi. *Design Patterns For Self Adaptive Systems Engineering*. International Journal of Software Engineering & Applications (IJSEA), Vol.6, No.4, pp. 11-26. July 2015

[20] Y. Brun et. al, *A Design Space for Self-Adaptive Systems*. Self-Adaptive Systems, Ed. Springer-Verlag Berlin Heidelberg, LNCS 7475, pp. 33-50, 2013

[21] M. Larsson, *An operation model for Microservices*. Technical report, 2015. [En línia]. https://callistaenterprise.se/blogg/teknik/2015/03/25/an-operations-model-for-microservices/ [Últim accés: 09/2020]

9. Annex

L'annex es divideix en dos apartats, un on es mostren els components que conformen la solució de µserveis amb les seues característiques i un altre on es mostra el *log* complet de tots els escenaris.

9.1. Components de la solució

En aquesta secció es mostraran els diversos components que conformen la solució. Cada taula contindrà les dades del component, on podrem veure la funció que realitza, quantes instàncies requereix per a funcionar, les restriccions que posseeix el component, les interfícies que requereix o ofereix i les connexions/connectors que posseeix.

SPLn - Servei Línia de Producció n (Production line)	
DESCRIPCIÓ FUNCIONAL:	Rep ordres de treball. Genera informació de producció (unitats fabricades, mostreig per qualitat, operaris participants, etc.) destinada principalment a l'ERP (ordres de treball de producció - PLWO), transformacions i Stock.
INSTÀNCIES:	Una per màquina física (4)
RESTRICCIONS:	Ha de situar-se físicament en el node computacional PL n (màquina / línia de producció n)
OFEREIX INTERFÍCIE:	Servei PL
REQUEREIX	Servei PLWO (1) Servei Transformacions (0..1) Servei Stock (1)
CONNEXIONS / CONNECTORS:	Servei ERP: bidireccional, es connecta a través d'un connector MQTT o REST

DevPLn - Dispositiu Línia de Producció n	
DESCRIPCIÓ FUNCIONAL:	Interfície física amb la línia de producció (màquina) núm. Permet extreure dades / indicadors físics de producció (velocitat de producció actual / real, parades de màquina, configuració de longitud de talls, nombre de talls realitzats, etc.) així com canviar la configuració de la màquina (velocitat de producció i longitud de tall). Genera informació sobre ubicació / localització de recursos que processa
INSTÀNCIES:	Una per màquina física (4)
RESTRICCIONS:	Ha de situar-se en el node PLN (màquina / línia de producció n)
OFEREIX INTERFÍCIE:	Servei DevicePL
REQUEREIX	Servei ThingsREG (1) Servei DigitalTwins (0..1) Servei Localitzacions (0..1)
CONNEXIONS / CONNECTORS:	- Servei ThingsREG: direccional, típicament a través d'MQTT / REST o Serveis Consul

TLPLn - Semàfor PLN	
DESCRIPCIÓ FUNCIONAL:	Proporciona feedback als operaris de la màquina PLN sobre el ritme de producció actual (en relació a l'esperat)
INSTÀNCIES:	Una per semàfor físic (4)
RESTRICCIONS:	Ha de situar-se en el node TLN (semàfor físic n)
OFEREIX INTERFÍCIE:	Servei TL
REQUEREIX	
CONNEXIONS / CONNECTORS:	

TLCPLn - Control de Semàfors PLN	
DESCRIPCIÓ FUNCIONAL:	Rep informació de l'ERP sobre la ordre de producció activa en la PLN, i d'acord a la velocitat actual (obtinguda directament de la DevPL o de l'DigitalTwins) controla el semàfor
INSTÀNCIES:	Una per semàfor físic (4)
RESTRICCIONS:	
OFEREIX INTERFÍCIE:	Servei TLCPL
REQUEREIX	Servei DevPL (1) Servei PLWO (1) Servei TL (*) Servei DigitalTwins (0..1)
CONNEXIONS / CONNECTORS:	

SREBn - Servei de rebobinadora n	
DESCRIPCIÓ FUNCIONAL:	Rep ordres de treball de rebobinat. Genera informació de producció (bobines processades / transformades) destinada principalment a l'ERP (ordres de treball de Rebobinatge - REBWO), transformacions, Stock.
INSTÀNCIES:	Una per màquina física (2)
RESTRICCIONS:	Ha de situar-se físicament en el node computacional REBn (rebobinadora n)
OFEREIX INTERFÍCIE:	Servei REB
REQUEREIX	Servei REBWO (1) Servei Transformacions (1) Servei Stock (1)
CONNEXIONS / CONNECTORS:	- Servei ERP: bidireccional, es connecta a través d'un connector MQTT o REST

DevREBn - Dispositiu rebobinadora n	
DESCRIPCIÓ FUNCIONAL:	Interfície física amb la rebobinadora (màquina) núm. Permet extreure dades / indicadors físics de producció (velocitat de rebobinat, metres generats, informació de talls) així com canviar la configuració de la màquina (velocitat de rebobinat, reinici de metres) Genera informació sobre ubicació / localització de recursos que processa
INSTÀNCIES:	Una per màquina física (2)
RESTRICCIONS:	Ha de situar-se en el node REBn (màquina / línia de producció n)
OFEREIX INTERFÍCIE:	Servei DeviceREB
REQUEREIX	Servei ThingsREG (1) Servei DigitalTwins (0..1) Servei Localitzacions (0..1)
CONNEXIONS / CONNECTORS:	- Servei ThingsREG: direccional, típicament a través d'MQTT / REST o Serveis Consul

ThingsREG - Servei de registre de Things	
DESCRIPCIÓ FUNCIONAL:	
INSTÀNCIES:	Escalable segons necessitat
RESTRICCIONS:	
OFEREIX INTERFÍCIE:	Servei ThingsREG
REQUEREIX	
CONNEXIONS / CONNECTORS:	

Things - Servei Digital Twins	
DESCRIPCIÓ FUNCIONAL:	Ofereix infraestructura IOT per a suport a el concepte de Digital Twins. Permet mantenir informació (shadow / digital twin) d'un objecte físic, encara que aquest no estigui disponible
INSTÀNCIES:	Escalable segons necessitat
RESTRICCIONS:	
OFEREIX INTERFÍCIE:	Servei DigitalTwins
REQUEREIX	Servei ThingsREG
CONNEXIONS / CONNECTORS:	

SLOC - Servei Localitzacions	
DESCRIPCIÓ FUNCIONAL:	Manté informació sobre la ubicació de diferents recursos 'localitzables' (operaris, màquines, bobines, etc.) dins de la fàbrica. Ofereix un model de localitzacions
INSTÀNCIES:	Escalable segons necessitat
RESTRICCIONS:	
OFEREIX INTERFÍCIE:	Servei Localitzacions Servei ModelLocaltizacions
REQUEREIX	
CONNEXIONS / CONNECTORS:	

SRS - Servei de Gestió de Recursos	
DESCRIPCIÓ FUNCIONAL:	Manté informació sobre identificadors i tipus de recursos de la fàbrica
INSTÀNCIES:	Escalable segons necessitat
RESTRICCIONS:	
OFEREIX INTERFÍCIE:	Servei Resources
REQUEREIX	
CONNEXIONS / CONNECTORS:	

ERP - ERP exposat com a servei	
DESCRIPCIÓ FUNCIONAL:	Manté informació sobre identificadors i tipus de recursos de la fàbrica
INSTÀNCIES:	una
RESTRICCIONS:	
OFEREIX INTERFÍCIE:	Ordres de Treball de PL - PLWO Ordres de Treball de REB - REBWO
REQUEREIX	
CONNEXIONS / CONNECTORS:	

STRANS - Servei de Transformacions	
DESCRIPCIÓ FUNCIONAL:	Ofereix suport per traçar les transformacions realitzades dins de la fàbrica
INSTÀNCIES:	Escalable segons necessitat
RESTRICCIONS:	
OFEREIX INTERFÍCIE:	Servei Transformacions
REQUEREIX	
CONNEXIONS / CONNECTORS:	

SSTOCK - Servei de Stock	
DESCRIPCIÓ FUNCIONAL:	Gestiona unitats disponibles tant de matèria primera bruta (bobines pretallades), en transformació (bobines tallades), o producte final (cantoneres produïdes). Els diferents processos productius o de logística (recepció / enviament de mercaderies) usen aquest servei.
INSTÀNCIES:	Escalable segons necessitat
RESTRICCIONS:	
OFEREIX INTERFÍCIE:	Servei Stock
REQUEREIX	
CONNEXIONS / CONNECTORS:	

9.2. Seqüències d'execució

En aquest apartat es mostren les seqüències d'execució dels diferents escenaris que hem proposat per fer les proves als prototips. Aquestes execucions generen *logs* d'informació que aporta informació per a entendre què està executant el sistema auto-gestionat, i quines operacions aplica sobre les configuracions del sistema.

9.2.1. Auto-configuració inicial

A continuació es mostra la seqüència d'execució completa per a la prova d'auto-configuració inicial que s'ha presentat a la secció 6.5.1 on demostrem que el sistema s'inicialitza correctament i de manera autònoma desplega un escenari inicial.

```
20200702222829057 1 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [Knowledge] BINDING SERVICE ...
20200702222829057 2 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [Knowledge] BINDING SERVICE ...
20200702222829080 6 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [Knowledge] SETTING PARAMETER ...
20200702222829080 8 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [Knowledge] DEPLOYING ...
20200702222829081 3 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [KnowledgegModule] BINDING SERVICE ...
20200702222829081 6 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [KnowledgegModule] GET SERVICE SUPPLY ...
20200702222829081 6 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [MonitoringModule] BINDING SERVICE ...
20200702222829081 9 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [KnowledgegModule] GET SERVICE SUPPLY ...
20200702222829082 1 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [AnalyzingModule] BINDING SERVICE ...
20200702222829082 7 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [AnalyzingModule] BINDING SERVICE ...
20200702222829082 8 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [AnalyzingModule] DEPLOYING ...
20200702222829082 8 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [AnalyzingModule] GET SERVICE SUPPLY ...
20200702222829082 8 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [MonitoringModule] BINDING SERVICE ...
20200702222829082 9 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [MonitoringModule] DEPLOYING ...
20200702222829083 3 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [KnowledgegModule] GET SERVICE SUPPLY ...
20200702222829083 4 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [PlanningModule] BINDING SERVICE ...
20200702222829083 4 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [PlanningModule] DEPLOYING ...
20200702222829083 5 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [PlanningModule] GET SERVICE SUPPLY ...
20200702222829083 5 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [AnalyzingModule] BINDING SERVICE ...
20200702222829083 9 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [KnowledgegModule] GET SERVICE SUPPLY ...
20200702222829084 0 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [ExecutingModule] BINDING SERVICE ...
20200702222829084 0 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [ExecutingModule] DEPLOYING ...
20200702222829084 0 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [ExecutingModule] GET SERVICE SUPPLY ...
20200702222829084 1 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [PlanningModule] BINDING SERVICE ...
20200702222829084 4 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [MonitoringModule] GET SERVICE SUPPLY ...
20200702222829084 4 [INFO ] ...ARC.impl.AdaptiveReadyComponent: [self-configure-monitor] BINDING SERVICE ..
20200702222829084 4 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [self-configure-monitor] DEPLOYING ...
20200702222829084 6 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [MonitoringModule] GET SERVICE SUPPLY ...
20200702222829084 7 [INFO ] ...ARC.impl.AdaptiveReadyComponent: [circuitbreaker-monitor] BINDING SERVICE ..
20200702222829084 7 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [circuitbreaker-monitor] DEPLOYING ...
20200702222829084 9 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [MonitoringModule] GET SERVICE SUPPLY ...
20200702222829084 9 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [loadbalancer-monitor] BINDING SERVICE ...
20200702222829084 9 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [loadbalancer-monitor] DEPLOYING ...
20200702222829085 1 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [MonitoringModule] GET SERVICE SUPPLY ...
20200702222829085 1 [INFO ] ..ARC.impl.AdaptiveReadyComponent: [host-load-level-monitor] BINDING SERVICE ..
20200702222829085 1 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [host-load-level-monitor] DEPLOYING ...
20200702222829086 0 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [AnalyzingModule] GET SERVICE SUPPLY ...
20200702222829086 0 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [SELF-CONFIGURE] BINDING SERVICE ...
20200702222829086 0 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [SELF-CONFIGURE] DEPLOYING ...
20200702222829086 2 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [AnalyzingModule] GET SERVICE SUPPLY ...
20200702222829086 3 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [circuitbreaker-configure-rule] BINDING SERVICE ...
20200702222829086 3 [INFO ] ..ARC.impl.AdaptiveReadyComponent: [circuitbreaker-configure-rule] DEPLOYING ..
20200702222829086 5 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [AnalyzingModule] GET SERVICE SUPPLY ...
20200702222829086 5 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [LOAD-BALANCER] BINDING SERVICE ...
20200702222829086 6 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [LOAD-BALANCER] DEPLOYING ...
20200702222829087 1 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [AnalyzingModule] GET SERVICE SUPPLY ...
20200702222829087 2 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [MOVE-CONTAINER] BINDING SERVICE ...
20200702222829087 2 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [MOVE-CONTAINER] DEPLOYING ...
20200702222829087 6 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [AnalyzingModule] GET SERVICE SUPPLY ...
20200702222829087 7 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [HOST-LOAD-LEVEL] BINDING SERVICE ...
20200702222829087 7 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [HOST-LOAD-LEVEL] DEPLOYING ...
20200702222829088 6 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [PlanningModule] BINDING SERVICE ...
20200702222829088 9 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [ExecutingModule] BINDING SERVICE ...
20200702222829090 0 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [ARM.SystemEffectors] DEPLOYING ...
20200702222829090 1 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [Executor] BINDING SERVICE ...
20200702222829090 9 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [self-configure-probe] BINDING SERVICE ...
20200702222829091 0 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [self-configure-probe] DEPLOYING ...
20200702222829091 2 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [circuitbreaker-configure-probe] BINDING SERVICE ...
20200702222829091 3 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [circuitbreaker-configure-probe] DEPLOYING ...
20200702222829091 5 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [loadbalancer-configure-probe] BINDING SERVICE ...
20200702222829091 6 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [loadbalancer-configure-probe] DEPLOYING ...
20200702222829091 8 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [host-load-level-probe] BINDING SERVICE ...
```

```
20200702222829091 8 [INFO ] ....ARC.impl.AdaptiveReadyComponent: [host-load-level-probe] DEPLOYING ...
20200702222829091 9 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: factory-reset
20200702222829092 0 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: factory-reset
20200702222829097 7 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule SELF-CONFIGURE> Asking for the next System Configuration
20200702222830009 7 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule SELF-CONFIGURE> Request Planning (c6835499c3a949879a63041429698788)
20200702222830010 2 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (c6835499c3a949879a63041429698788) <<<
20200702222830037 7 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DEPLOYING HOST
20200702222830038 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] DEPLOYING HOST
20200702222830038 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL3] DEPLOYING HOST
20200702222830038 5 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL4] DEPLOYING HOST
20200702222830038 6 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW1] DEPLOYING HOST
20200702222830038 7 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW2] DEPLOYING HOST
20200702222830038 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL1] DEPLOYING HOST
20200702222830038 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] DEPLOYING HOST
20200702222830038 9 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DEPLOYING HOST
20200702222830039 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL4] DEPLOYING HOST
20200702222830039 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DEPLOYING HOST
20200702222830039 2 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] EXPOSING EDNPOINT 18000
20200702222830039 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] EXPOSING EDNPOINT 18001
20200702222830039 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] EXPOSING EDNPOINT 18000
20200702222830039 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] EXPOSING EDNPOINT 18001
20200702222830039 4 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL3] EXPOSING EDNPOINT 18000
20200702222830039 4 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL3] EXPOSING EDNPOINT 18001
20200702222830039 4 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL4] EXPOSING EDNPOINT 18000
20200702222830039 5 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL4] EXPOSING EDNPOINT 18001
20200702222830039 5 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW1] EXPOSING EDNPOINT 18010
20200702222830039 5 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW1] EXPOSING EDNPOINT 18011
20200702222830039 5 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW2] EXPOSING EDNPOINT 18010
20200702222830039 6 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW2] EXPOSING EDNPOINT 18011
20200702222830039 7 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL1] EXPOSING EDNPOINT 18021
20200702222830039 7 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL1] EXPOSING EDNPOINT 18020
20200702222830039 7 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] EXPOSING EDNPOINT 18021
20200702222830039 7 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] EXPOSING EDNPOINT 18020
20200702222830039 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] EXPOSING EDNPOINT 18021
20200702222830039 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] EXPOSING EDNPOINT 18020
20200702222830039 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL4] EXPOSING EDNPOINT 18021
20200702222830039 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL4] EXPOSING EDNPOINT 18020
20200702222830039 9 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] EXPOSING EDNPOINT 18030
20200702222830039 9 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] EXPOSING EDNPOINT 18031
20200702222830039 9 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] EXPOSING EDNPOINT 18040
20200702222830040 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] EXPOSING EDNPOINT 18041
20200702222830040 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] EXPOSING EDNPOINT 18050
20200702222830040 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] EXPOSING EDNPOINT 18060
20200702222830040 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] EXPOSING EDNPOINT 18070
20200702222830040 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] EXPOSING EDNPOINT 18080
20200702222830040 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] EXPOSING EDNPOINT 18081
20200702222830040 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DEPLOYING CONTAINER CPL1
20200702222830040 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL1] DEPLOYING CONTAINER CPL1 IN HOST HostPL1
20200702222830040 9 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] DEPLOYING CONTAINER CPL2
20200702222830041 1 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL2] DEPLOYING CONTAINER CPL2 IN HOST HostPL2
20200702222830041 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL3] DEPLOYING CONTAINER CPL3
20200702222830041 2 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL3] DEPLOYING CONTAINER CPL3 IN HOST HostPL3
20200702222830041 2 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL4] DEPLOYING CONTAINER CPL4
20200702222830041 3 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL4] DEPLOYING CONTAINER CPL4 IN HOST HostPL4
20200702222830041 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DEPLOYING CONTAINER CDevPL1
20200702222830041 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL1] DEPLOYING CONTAINER CDevPL1 IN HOST HostPL1
20200702222830041 9 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] DEPLOYING CONTAINER CDevPL2
20200702222830041 9 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL2] DEPLOYING CONTAINER CDevPL2 IN HOST HostPL2
20200702222830042 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL3] DEPLOYING CONTAINER CDevPL3
20200702222830042 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL3] DEPLOYING CONTAINER CDevPL3 IN HOST HostPL3
20200702222830042 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL4] DEPLOYING CONTAINER CDevPL4
20200702222830042 3 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL4] DEPLOYING CONTAINER CDevPL4 IN HOST HostPL4
20200702222830042 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW1] DEPLOYING CONTAINER CREW1
20200702222830042 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CREW1] DEPLOYING CONTAINER CREW1 IN HOST HostREW1
20200702222830042 6 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW2] DEPLOYING CONTAINER CREW2
20200702222830042 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CREW2] DEPLOYING CONTAINER CREW2 IN HOST HostREW2
20200702222830042 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW1] DEPLOYING CONTAINER CDevREW1
20200702222830042 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevREW1] DEPLOYING CONTAINER CDevREW1 IN HOST HostREW1
20200702222830042 9 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW2] DEPLOYING CONTAINER CDevREW2
20200702222830042 9 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevREW2] DEPLOYING CONTAINER CDevREW2 IN HOST HostREW2
20200702222830043 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL1] DEPLOYING CONTAINER CDevTLPL1
20200702222830043 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL1] DEPLOYING CONTAINER CDevTLPL1 IN HOST HostTL1
20200702222830043 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] DEPLOYING CONTAINER CDevTLPL2
20200702222830043 1 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL2] DEPLOYING CONTAINER CDevTLPL2 IN HOST HostTL2
20200702222830043 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DEPLOYING CONTAINER CDevTLPL3
20200702222830043 2 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL3] DEPLOYING CONTAINER CDevTLPL3 IN HOST HostTL3
20200702222830043 2 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL4] DEPLOYING CONTAINER CDevTLPL4
20200702222830043 3 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL4] DEPLOYING CONTAINER CDevTLPL4 IN HOST HostTL4
20200702222830043 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL1] DEPLOYING CONTAINER CTLC1
20200702222830044 1 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC1] DEPLOYING CONTAINER CTLC1 IN HOST HostTL1
20200702222830044 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] DEPLOYING CONTAINER CTLC2
20200702222830044 2 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC2] DEPLOYING CONTAINER CTLC2 IN HOST HostTL2
20200702222830044 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DEPLOYING CONTAINER CTLC3
20200702222830044 3 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC3] DEPLOYING CONTAINER CTLC3 IN HOST HostTL3
20200702222830044 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL4] DEPLOYING CONTAINER CTLC4
20200702222830044 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC4] DEPLOYING CONTAINER CTLC4 IN HOST HostTL4
20200702222830044 4 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DEPLOYING CONTAINER CERP
20200702222830044 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CERP] DEPLOYING CONTAINER CERP IN HOST HostERP
20200702222830044 5 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DEPLOYING CONTAINER CSLOC
20200702222830044 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CSLOC] DEPLOYING CONTAINER CSLOC IN HOST HostERP
20200702222830044 6 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DEPLOYING CONTAINER CResources
20200702222830044 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CResources] DEPLOYING CONTAINER CResources IN HOST HostERP
20200702222830044 6 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DEPLOYING CONTAINER CTransf
20200702222830044 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTransf] DEPLOYING CONTAINER CTransf IN HOST HostERP
20200702222830044 7 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DEPLOYING CONTAINER CStock
20200702222830044 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CStock] DEPLOYING CONTAINER CStock IN HOST HostERP
20200702222830044 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DEPLOYING CONTAINER CThings
20200702222830044 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] DEPLOYING CONTAINER CThings IN HOST HostERP
20200702222830044 9 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DEPLOYING CONTAINER CThingsReg
20200702222830044 9 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThingsReg] DEPLOYING CONTAINER CThingsReg IN HOST HostERP
20200702222830045 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DELEGATE ENDPOINT 18000 TO CONTAINER CPL1 ENDPOINT 8080
20200702222830045 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DELEGATE ENDPOINT 18001 TO CONTAINER CDevPL1 ENDPOINT 8081
20200702222830045 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] DELEGATE ENDPOINT 18000 TO CONTAINER CPL2 ENDPOINT 8080
20200702222830045 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] DELEGATE ENDPOINT 18001 TO CONTAINER CDevPL2 ENDPOINT 8081
20200702222830045 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL3] DELEGATE ENDPOINT 18000 TO CONTAINER CPL3 ENDPOINT 8080
20200702222830045 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL3] DELEGATE ENDPOINT 18001 TO CONTAINER CDevPL3 ENDPOINT 8081
20200702222830045 6 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL4] DELEGATE ENDPOINT 18000 TO CONTAINER CPL4 ENDPOINT 8080
20200702222830045 7 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL4] DELEGATE ENDPOINT 18001 TO CONTAINER CDevPL4 ENDPOINT 8081
```

```
20200702222830045 7 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW1] DELEGATE ENDPOINT 18010 TO CONTAINER CREW1 ENDPOINT 8082
20200702222830045 7 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW1] DELEGATE ENDPOINT 18011 TO CONTAINER CDevREW1 ENDPOINT 8083
20200702222830045 7 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW2] DELEGATE ENDPOINT 18010 TO CONTAINER CREW2 ENDPOINT 8082
20200702222830045 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW2] DELEGATE ENDPOINT 18011 TO CONTAINER CDevREW2 ENDPOINT 8083
20200702222830045 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL1] DELEGATE ENDPOINT 18021 TO CONTAINER CDevTLPL1 ENDPOINT 8085
20200702222830045 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL1] DELEGATE ENDPOINT 18020 TO CONTAINER CTLC1 ENDPOINT 8084
20200702222830045 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] DELEGATE ENDPOINT 18021 TO CONTAINER CDevTLPL2 ENDPOINT 8085
20200702222830046 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] DELEGATE ENDPOINT 18020 TO CONTAINER CTLC2 ENDPOINT 8084
20200702222830046 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DELEGATE ENDPOINT 18021 TO CONTAINER CDevTLPL3 ENDPOINT 8085
20200702222830046 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DELEGATE ENDPOINT 18020 TO CONTAINER CTLC3 ENDPOINT 8084
20200702222830046 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL4] DELEGATE ENDPOINT 18021 TO CONTAINER CDevTLPL4 ENDPOINT 8085
20200702222830046 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL4] DELEGATE ENDPOINT 18020 TO CONTAINER CTLC4 ENDPOINT 8084
20200702222830046 2 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DELEGATE ENDPOINT 18030 TO CONTAINER CERP ENDPOINT 8086
20200702222830046 2 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DELEGATE ENDPOINT 18031 TO CONTAINER CERP ENDPOINT 8087
20200702222830046 2 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DELEGATE ENDPOINT 18040 TO CONTAINER CSLOC ENDPOINT 8088
20200702222830046 2 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DELEGATE ENDPOINT 18041 TO CONTAINER CSLOC ENDPOINT 8089
20200702222830046 2 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DELEGATE ENDPOINT 18050 TO CONTAINER CResources ENDPOINT 8090
20200702222830046 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DELEGATE ENDPOINT 18060 TO CONTAINER CTransf ENDPOINT 8091
20200702222830046 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DELEGATE ENDPOINT 18070 TO CONTAINER CStock ENDPOINT 8092
20200702222830046 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DELEGATE ENDPOINT 18080 TO CONTAINER CThings ENDPOINT 8093
20200702222830046 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DELEGATE ENDPOINT 18081 TO CONTAINER CThingsReg ENDPOINT 8094
20200702222830046 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL1] EXPOSING ENDPOINT 8080
20200702222830046 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL2] EXPOSING ENDPOINT 8080
20200702222830046 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL3] EXPOSING ENDPOINT 8080
20200702222830046 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL4] EXPOSING ENDPOINT 8080
20200702222830046 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL1] EXPOSING ENDPOINT 8081
20200702222830046 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL2] EXPOSING ENDPOINT 8081
20200702222830046 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL3] EXPOSING ENDPOINT 8081
20200702222830046 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL4] EXPOSING ENDPOINT 8081
20200702222830046 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CREW1] EXPOSING ENDPOINT 8082
20200702222830046 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CREW2] EXPOSING ENDPOINT 8082
20200702222830046 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevREW1] EXPOSING ENDPOINT 8083
20200702222830046 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevREW2] EXPOSING ENDPOINT 8083
20200702222830046 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL1] EXPOSING ENDPOINT 8085
20200702222830046 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL2] EXPOSING ENDPOINT 8085
20200702222830046 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL3] EXPOSING ENDPOINT 8085
20200702222830046 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL4] EXPOSING ENDPOINT 8085
20200702222830046 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC1] EXPOSING ENDPOINT 8084
20200702222830046 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC2] EXPOSING ENDPOINT 8084
20200702222830046 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC3] EXPOSING ENDPOINT 8084
20200702222830046 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC4] EXPOSING ENDPOINT 8084
20200702222830046 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CERP] EXPOSING ENDPOINT 8086
20200702222830047 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CERP] EXPOSING ENDPOINT 8087
20200702222830047 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CSLOC] EXPOSING ENDPOINT 8088
20200702222830047 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CSLOC] EXPOSING ENDPOINT 8089
20200702222830047 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CResources] EXPOSING ENDPOINT 8090
20200702222830047 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTransf] EXPOSING ENDPOINT 8091
20200702222830047 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CStock] EXPOSING ENDPOINT 8092
20200702222830047 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] EXPOSING ENDPOINT 8093
20200702222830047 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] EXPOSING ENDPOINT 8094
20200702222830047 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] EXPOSING ENDPOINT 8095
20200702222830047 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThingsReg] EXPOSING ENDPOINT 8094
20200702222830047 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL1] DEPLOYING MICROSERVICE PL1
20200702222830048 5 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1] DEPLOYING MICROSERVICE PL1 IN CONTAINER CPL1
20200702222830062 9 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL2] DEPLOYING MICROSERVICE PL2
20200702222830063 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2] DEPLOYING MICROSERVICE PL2 IN CONTAINER CPL2
20200702222830063 2 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL3] DEPLOYING MICROSERVICE PL3
20200702222830063 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3] DEPLOYING MICROSERVICE PL3 IN CONTAINER CPL3
20200702222830063 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL4] DEPLOYING MICROSERVICE PL4
20200702222830063 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL4] DEPLOYING MICROSERVICE PL4 IN CONTAINER CPL4
20200702222830063 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL1] DEPLOYING MICROSERVICE DevPL1
20200702222830063 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL1] DEPLOYING MICROSERVICE DevPL1 IN CONTAINER CDevPL1
20200702222830063 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL2] DEPLOYING MICROSERVICE DevPL2
20200702222830063 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL2] DEPLOYING MICROSERVICE DevPL2 IN CONTAINER CDevPL2
20200702222830063 9 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL3] DEPLOYING MICROSERVICE DevPL3
20200702222830064 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL3] DEPLOYING MICROSERVICE DevPL3 IN CONTAINER CDevPL3
20200702222830064 1 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL4] DEPLOYING MICROSERVICE DevPL4
20200702222830064 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL4] DEPLOYING MICROSERVICE DevPL4 IN CONTAINER CDevPL4
20200702222830064 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CREW1] DEPLOYING MICROSERVICE REW1
20200702222830064 5 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW1] DEPLOYING MICROSERVICE REW1 IN CONTAINER CREW1
20200702222830064 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CREW2] DEPLOYING MICROSERVICE REW2
20200702222830064 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW2] DEPLOYING MICROSERVICE REW2 IN CONTAINER CREW2
20200702222830064 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevREW1] DEPLOYING MICROSERVICE DevREW1
20200702222830064 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW1] DEPLOYING MICROSERVICE DevREW1 IN CONTAINER CDevREW1
20200702222830064 9 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevREW2] DEPLOYING MICROSERVICE DevREW2
20200702222830065 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW2] DEPLOYING MICROSERVICE DevREW2 IN CONTAINER CDevREW2
20200702222830065 1 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL1] DEPLOYING MICROSERVICE DevTLPL1
20200702222830065 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL1] DEPLOYING MICROSERVICE DevTLPL1 IN CONTAINER CDevTLPL1
20200702222830065 2 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL2] DEPLOYING MICROSERVICE DevTLPL2
20200702222830065 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2] DEPLOYING MICROSERVICE DevTLPL2 IN CONTAINER CDevTLPL2
20200702222830065 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL3] DEPLOYING MICROSERVICE DevTLPL3
20200702222830065 5 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL3] DEPLOYING MICROSERVICE DevTLPL3 IN CONTAINER CDevTLPL3
20200702222830065 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL4] DEPLOYING MICROSERVICE DevTLPL4
20200702222830065 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4] DEPLOYING MICROSERVICE DevTLPL4 IN CONTAINER CDevTLPL4
20200702222830065 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC1] DEPLOYING MICROSERVICE TLC1
20200702222830067 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1] DEPLOYING MICROSERVICE TLC1 IN CONTAINER CTLC1
20200702222830067 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC2] DEPLOYING MICROSERVICE TLC2
20200702222830067 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2] DEPLOYING MICROSERVICE TLC2 IN CONTAINER CTLC2
20200702222830068 1 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC3] DEPLOYING MICROSERVICE TLC3
20200702222830068 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3] DEPLOYING MICROSERVICE TLC3 IN CONTAINER CTLC3
20200702222830068 3 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC4] DEPLOYING MICROSERVICE TLC4
20200702222830068 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4] DEPLOYING MICROSERVICE TLC4 IN CONTAINER CTLC4
20200702222830068 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CERP] DEPLOYING MICROSERVICE ERP
20200702222830068 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [ERP] DEPLOYING MICROSERVICE ERP IN CONTAINER CERP
20200702222830068 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CSLOC] DEPLOYING MICROSERVICE SLOC
20200702222830068 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [SLOC] DEPLOYING MICROSERVICE SLOC IN CONTAINER CSLOC
20200702222830069 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CResources] DEPLOYING MICROSERVICE Resources
20200702222830069 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [Resources] DEPLOYING MICROSERVICE Resources IN CONTAINER CResources
20200702222830069 2 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTransf] DEPLOYING MICROSERVICE Transf
20200702222830069 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [Transf] DEPLOYING MICROSERVICE Transf IN CONTAINER CTransf
20200702222830069 3 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CStock] DEPLOYING MICROSERVICE Stock
20200702222830069 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [Stock] DEPLOYING MICROSERVICE Stock IN CONTAINER CStock
20200702222830069 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] DEPLOYING MICROSERVICE Things
20200702222830069 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [Things] DEPLOYING MICROSERVICE Things IN CONTAINER CThings
20200702222830069 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] DEPLOYING MICROSERVICE ThingsProva
20200702222830069 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [ThingsProva] DEPLOYING MICROSERVICE ThingsProva IN CONTAINER CThings
20200702222830070 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] DEPLOYING MICROSERVICE ThingsTest
20200702222830070 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [ThingsTest] DEPLOYING MICROSERVICE ThingsTest IN CONTAINER CThings
```

```
20200702222830070 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThingsReg] DEPLOYING MICROSERVICE ThingsReg
20200702222830070 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [ThingsReg] DEPLOYING MICROSERVICE ThingsReg IN CONTAINER CThingsReg
20200702222830071 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702222830071 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702222830071 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702222830071 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL4] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702222830071 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL1] EXPOSING SERVICE DevPL THROUGH INTERFACE devpl
20200702222830071 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL2] EXPOSING SERVICE DevPL THROUGH INTERFACE devpl
20200702222830072 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL3] EXPOSING SERVICE DevPL THROUGH INTERFACE devpl
20200702222830072 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL4] EXPOSING SERVICE DevPL THROUGH INTERFACE devpl
20200702222830072 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW1] EXPOSING SERVICE REW THROUGH INTERFACE rew
20200702222830072 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW2] EXPOSING SERVICE REW THROUGH INTERFACE rew
20200702222830073 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW1] EXPOSING SERVICE DevREW THROUGH INTERFACE devrew
20200702222830073 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW2] EXPOSING SERVICE DevREW THROUGH INTERFACE devrew
20200702222830073 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL1] EXPOSING SERVICE DevTL THROUGH INTERFACE devtl
20200702222830073 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2] EXPOSING SERVICE DevTL THROUGH INTERFACE devtl
20200702222830073 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL3] EXPOSING SERVICE DevTL THROUGH INTERFACE devtl
20200702222830073 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4] EXPOSING SERVICE DevTL THROUGH INTERFACE devtl
20200702222830073 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1] EXPOSING SERVICE TLC THROUGH INTERFACE tlcpl
20200702222830073 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2] EXPOSING SERVICE TLC THROUGH INTERFACE tlcpl
20200702222830074 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3] EXPOSING SERVICE TLC THROUGH INTERFACE tlcpl
20200702222830074 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4] EXPOSING SERVICE TLC THROUGH INTERFACE tlcpl
20200702222830074 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [ERP] EXPOSING SERVICE PLWO THROUGH INTERFACE plwo
20200702222830074 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [ERP] EXPOSING SERVICE REWWO THROUGH INTERFACE rewwo
20200702222830074 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [SLOC] EXPOSING SERVICE Locations THROUGH INTERFACE loc
20200702222830074 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [SLOC] EXPOSING SERVICE LocationsModel THROUGH INTERFACE loc-model
20200702222830074 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [Resources] EXPOSING SERVICE Resources THROUGH INTERFACE res
20200702222830074 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [Transf] EXPOSING SERVICE Transf THROUGH INTERFACE transf
20200702222830074 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [Stock] EXPOSING SERVICE Stock THROUGH INTERFACE stock
20200702222830074 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [Things] EXPOSING SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702222830074 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [ThingsProva] EXPOSING SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702222830074 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [ThingsTest] EXPOSING SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702222830074 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [ThingsReg] EXPOSING SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702222830074 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702222830074 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702222830074 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702222830074 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702222830074 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702222830074 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702222830074 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702222830074 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702222830074 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702222830074 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL4] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702222830074 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL4] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702222830074 5 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL4] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702222830074 5 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL1] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702222830074 5 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL1] DEFINING REQUIRED SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702222830074 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL1] DEFINING REQUIRED SERVICE Locations THROUGH INTERFACE loc
20200702222830074 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL2] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702222830074 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL2] DEFINING REQUIRED SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702222830074 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL2] DEFINING REQUIRED SERVICE Locations THROUGH INTERFACE loc
20200702222830074 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL3] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702222830074 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL3] DEFINING REQUIRED SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702222830074 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL3] DEFINING REQUIRED SERVICE Locations THROUGH INTERFACE loc
20200702222830074 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL4] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702222830074 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL4] DEFINING REQUIRED SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702222830074 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL4] DEFINING REQUIRED SERVICE Locations THROUGH INTERFACE loc
20200702222830074 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW1] DEFINING REQUIRED SERVICE REWWO THROUGH INTERFACE rewwo
20200702222830074 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW1] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702222830074 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW1] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702222830074 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW2] DEFINING REQUIRED SERVICE REWWO THROUGH INTERFACE rewwo
20200702222830074 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW2] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702222830075 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW2] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702222830075 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW1] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702222830075 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW1] DEFINING REQUIRED SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702222830075 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW1] DEFINING REQUIRED SERVICE Locations THROUGH INTERFACE loc
20200702222830075 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW2] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702222830075 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW2] DEFINING REQUIRED SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702222830075 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW2] DEFINING REQUIRED SERVICE Locations THROUGH INTERFACE loc
20200702222830075 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL1] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702222830075 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL1] DEFINING REQUIRED SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702222830075 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702222830075 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2] DEFINING REQUIRED SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702222830075 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL3] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702222830075 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL3] DEFINING REQUIRED SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702222830075 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702222830075 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4] DEFINING REQUIRED SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702222830075 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1] DEFINING REQUIRED SERVICE DevPL THROUGH INTERFACE devpl
20200702222830075 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1] DEFINING REQUIRED SERVICE DevTL THROUGH INTERFACE devtl
20200702222830075 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702222830075 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702222830075 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2] DEFINING REQUIRED SERVICE DevPL THROUGH INTERFACE devpl
20200702222830075 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2] DEFINING REQUIRED SERVICE DevTL THROUGH INTERFACE devtl
20200702222830075 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702222830075 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702222830075 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3] DEFINING REQUIRED SERVICE DevPL THROUGH INTERFACE devpl
20200702222830075 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3] DEFINING REQUIRED SERVICE DevTL THROUGH INTERFACE devtl
20200702222830075 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702222830075 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702222830075 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4] DEFINING REQUIRED SERVICE DevPL THROUGH INTERFACE devpl
20200702222830075 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4] DEFINING REQUIRED SERVICE DevTL THROUGH INTERFACE devtl
20200702222830075 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702222830075 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702222830075 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [Things] DEFINING REQUIRED SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702222830075 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [ThingsProva] DEFINING REQUIRED SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702222830075 5 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [ThingsTest] DEFINING REQUIRED SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702222830075 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL1] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL1 ENDPOINT pl
20200702222830075 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL2] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL2 ENDPOINT pl
20200702222830075 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL3] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL3 ENDPOINT pl
20200702222830075 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL4] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL4 ENDPOINT pl
20200702222830075 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL1] DELEGATE ENDPOINT 8081 TO MICROSERVICE DevPL1 ENDPOINT devpl
20200702222830075 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL2] DELEGATE ENDPOINT 8081 TO MICROSERVICE DevPL2 ENDPOINT devpl
20200702222830075 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL3] DELEGATE ENDPOINT 8081 TO MICROSERVICE DevPL3 ENDPOINT devpl
20200702222830075 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL4] DELEGATE ENDPOINT 8081 TO MICROSERVICE DevPL4 ENDPOINT devpl
20200702222830075 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CREW1] DELEGATE ENDPOINT 8082 TO MICROSERVICE REW1 ENDPOINT rew
20200702222830075 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CREW2] DELEGATE ENDPOINT 8082 TO MICROSERVICE REW2 ENDPOINT rew
20200702222830075 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevREW1] DELEGATE ENDPOINT 8083 TO MICROSERVICE DevREW1 ENDPOINT devrew
20200702222830075 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevREW2] DELEGATE ENDPOINT 8083 TO MICROSERVICE DevREW2 ENDPOINT devrew
```

```
20200702222830758 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL1] DELEGATE ENDPOINT 8085 TO MICROSERVICE DevTLPL1 ENDPOINT devtl
20200702222830758 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL2] DELEGATE ENDPOINT 8085 TO MICROSERVICE DevTLPL2 ENDPOINT devtl
20200702222830758 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL3] DELEGATE ENDPOINT 8085 TO MICROSERVICE DevTLPL3 ENDPOINT devtl
20200702222830758 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL4] DELEGATE ENDPOINT 8085 TO MICROSERVICE DevTLPL4 ENDPOINT devtl
20200702222830759 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC1] DELEGATE ENDPOINT 8084 TO MICROSERVICE TLC1 ENDPOINT tlcpl
20200702222830759 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC2] DELEGATE ENDPOINT 8084 TO MICROSERVICE TLC2 ENDPOINT tlcpl
20200702222830759 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC3] DELEGATE ENDPOINT 8084 TO MICROSERVICE TLC3 ENDPOINT tlcpl
20200702222830759 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC4] DELEGATE ENDPOINT 8084 TO MICROSERVICE TLC4 ENDPOINT tlcpl
20200702222830759 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CERP] DELEGATE ENDPOINT 8086 TO MICROSERVICE ERP ENDPOINT plwo
20200702222830759 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CERP] DELEGATE ENDPOINT 8087 TO MICROSERVICE ERP ENDPOINT rewwo
20200702222830759 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CSLOC] DELEGATE ENDPOINT 8088 TO MICROSERVICE SLOC ENDPOINT loc
20200702222830760 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CSLOC] DELEGATE ENDPOINT 8089 TO MICROSERVICE SLOC ENDPOINT loc-model
20200702222830760 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CResources] DELEGATE ENDPOINT 8090 TO MICROSERVICE Resources ENDPOINT res
20200702222830760 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTransf] DELEGATE ENDPOINT 8091 TO MICROSERVICE Transf ENDPOINT transf
20200702222830760 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CStock] DELEGATE ENDPOINT 8092 TO MICROSERVICE Stock ENDPOINT stock
20200702222830760 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] DELEGATE ENDPOINT 8093 TO MICROSERVICE Things ENDPOINT digTwins
20200702222830760 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] DELEGATE ENDPOINT 8094 TO MICROSERVICE ThingsProva ENDPOINT digTwins
20200702222830760 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] DELEGATE ENDPOINT 8095 TO MICROSERVICE ThingsTest ENDPOINT digTwins
20200702222830761 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThingsReg] DELEGATE ENDPOINT 8094 TO MICROSERVICE ThingsReg ENDPOINT thingsReg
20200702222830761 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702222830761 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702222830761 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702222830761 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702222830762 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702222830762 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702222830762 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702222830762 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702222830762 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702222830762 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL4] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702222830763 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL4] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702222830779 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL4] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702222830780 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL1] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702222830780 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL1] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702222830780 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL1] PUBLIC BINDING loc TO HOST HostERP ENDPOINT 18040
20200702222830780 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL2] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702222830781 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL2] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702222830781 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL2] PUBLIC BINDING loc TO HOST HostERP ENDPOINT 18040
20200702222830781 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL3] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702222830781 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL3] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702222830781 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL3] PUBLIC BINDING loc TO HOST HostERP ENDPOINT 18040
20200702222830781 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL4] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702222830781 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL4] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702222830782 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL4] PUBLIC BINDING loc TO HOST HostERP ENDPOINT 18040
20200702222830782 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW1] PUBLIC BINDING rewwo TO HOST HostERP ENDPOINT 18031
20200702222830782 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW1] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702222830782 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW1] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702222830782 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW2] PUBLIC BINDING rewwo TO HOST HostERP ENDPOINT 18031
20200702222830782 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW2] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702222830782 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW2] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702222830782 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW1] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702222830783 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW1] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702222830783 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW1] PUBLIC BINDING loc TO HOST HostERP ENDPOINT 18040
20200702222830783 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW2] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702222830783 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW2] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702222830783 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW2] PUBLIC BINDING loc TO HOST HostERP ENDPOINT 18040
20200702222830783 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL1] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702222830783 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL1] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702222830785 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702222830785 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702222830787 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL3] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702222830787 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL3] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702222830788 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702222830788 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702222830788 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702222830788 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702222830788 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1] PUBLIC BINDING devtl TO HOST HostTL1 ENDPOINT 18021
20200702222830788 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1] PUBLIC BINDING devpl TO HOST HostPL1 ENDPOINT 18001
20200702222830788 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702222830788 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702222830789 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2] PUBLIC BINDING devtl TO HOST HostTL2 ENDPOINT 18021
20200702222830789 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2] PUBLIC BINDING devpl TO HOST HostPL2 ENDPOINT 18001
20200702222830789 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702222830789 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702222830789 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3] PUBLIC BINDING devtl TO HOST HostTL3 ENDPOINT 18021
20200702222830790 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3] PUBLIC BINDING devpl TO HOST HostPL3 ENDPOINT 18001
20200702222830790 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702222830790 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702222830790 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4] PUBLIC BINDING devtl TO HOST HostTL4 ENDPOINT 18021
20200702222830790 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4] PUBLIC BINDING devpl TO HOST HostPL4 ENDPOINT 18001
20200702222830791 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [Things] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702222830791 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [ThingsProva] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702222830791 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [ThingsTest] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702222830791 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702222830960 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (c6835499c3a949879a63041429698788) <<<
```

9.2.2. Load Balancer i del Circuit Breaker

A continuació es mostra la seqüència d'execució completa per a la prova de les adaptacions "*Load Balancer*" i "*Circuit Breaker*" que s'han presentat a la secció 6.5.2 on demostrem que el sistema pot realitzar aquestes funcions i incloure els patrons autònomament per corregir possibles anomalies.

```
osgi> lb PL1 high
20200702223041067 6 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: PL1 => high
20200702223041067 9 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: PL1 => high
20200702223041068 1 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule LOAD-BALANCER> Asking for the next System Configuration
20200702223041069 4 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule LOAD-BALANCER> Request Planning (2de0aafe1f4148aab1947a12ea5e97b0)
20200702223041069 5 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (2de0aafe1f4148aab1947a12ea5e97b0) <<<
20200702223041076 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DELEGATING CONTAINER PL1-lb
20200702223041076 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-lb] DEPLOYING CONTAINER PL1-lb IN HOST HostPL1
20200702223041076 6 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW1] DEPLOYING CONTAINER PL1-1
20200702223041076 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-1] DEPLOYING CONTAINER PL1-1 IN HOST HostREW1
20200702223041076 7 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL4] DEPLOYING CONTAINER PL1-2
20200702223041076 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-2] DEPLOYING CONTAINER PL1-2 IN HOST HostPL4
20200702223041076 9 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL4] DEPLOYING CONTAINER PL1-3
20200702223041076 9 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-3] DEPLOYING CONTAINER PL1-3 IN HOST HostTL4
20200702223041077 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DELEGATE ENDPOINT 18000 TO CONTAINER PL1-lb ENDPOINT 8080
20200702223041077 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-lb] EXPOSING ENDPOINT 8080
20200702223041077 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-1] EXPOSING ENDPOINT 8080
20200702223041077 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-2] EXPOSING ENDPOINT 8080
20200702223041077 1 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-3] EXPOSING ENDPOINT 8080
20200702223041077 1 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-1] DEPLOYING MICROSERVICE PL1-1
20200702223041077 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-1] DEPLOYING MICROSERVICE PL1-1 IN CONTAINER PL1-1
20200702223041077 2 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-2] DEPLOYING MICROSERVICE PL1-2
20200702223041077 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-2] DEPLOYING MICROSERVICE PL1-2 IN CONTAINER PL1-2
20200702223041077 3 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-3] DEPLOYING MICROSERVICE PL1-3
20200702223041077 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-3] DEPLOYING MICROSERVICE PL1-3 IN CONTAINER PL1-3
20200702223041077 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-lb] DEPLOYING MICROSERVICE PL1-lb
20200702223041077 5 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-lb] DEPLOYING MICROSERVICE PL1-lb IN CONTAINER PL1-lb
20200702223041077 5 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-1] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702223041077 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-2] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702223041077 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-3] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702223041077 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-lb] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702223041077 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-1] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702223041077 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-1] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702223041077 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-1] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702223041077 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-2] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702223041077 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-2] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702223041077 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-2] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702223041077 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-3] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702223041077 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-3] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702223041077 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-3] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702223041077 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-lb] DEFINING REQUIRED SERVICE PL THROUGH INTERFACE reqpl
20200702223041077 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-lb] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL1-lb ENDPOINT pl
20200702223041077 9 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-1] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL1-1 ENDPOINT pl
20200702223041077 9 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-2] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL1-2 ENDPOINT pl
20200702223041077 9 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL1-3] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL1-3 ENDPOINT pl
20200702223041077 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-1] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702223041078 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-1] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702223041078 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-1] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702223041078 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-2] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702223041078 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-2] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702223041078 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-2] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702223041078 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-3] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702223041078 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-3] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702223041078 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1-3] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702223041078 2 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702223041086 7 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (2de0aafe1f4148aab1947a12ea5e97b0) <<<

osgi> cb PL2 high

20200702223125076 7 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostPL2)[Container: CPL2]<Microservice: PL2> => high
20200702223125076 8 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostPL2)[Container: CPL2]<Microservice: PL2> => high
20200702223125076 9 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule circuitbreaker-configure-rule> Asking for the next System Configuration
20200702223125077 5 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule circuitbreaker-configure-rule> Request Planning (f918ff0fb7f1426e99cbce10fd2fef11)
20200702223125077 6 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (f918ff0fb7f1426e99cbce10fd2fef11) <<<
20200702223125083 5 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DEPLOYING CONTAINER PL2-cb
20200702223125083 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL2-cb] DEPLOYING CONTAINER PL2-cb IN HOST HostPL1
20200702223125083 6 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DELEGATE ENDPOINT 18000 TO CONTAINER PL2-cb ENDPOINT 8080
20200702223125083 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL2-cb] EXPOSING ENDPOINT 8080
20200702223125083 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL2-cb] DEPLOYING MICROSERVICE PL2-cb
20200702223125083 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-cb] DEPLOYING MICROSERVICE PL2-cb IN CONTAINER PL2-cb
20200702223125083 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-cb] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702223125083 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-cb] DEFINING REQUIRED SERVICE PL THROUGH INTERFACE reqpl
20200702223125083 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL2-cb] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL2-cb ENDPOINT pl
20200702223125083 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-cb] PUBLIC BINDING reqpl TO HOST HostPL2 ENDPOINT 18000
20200702223125083 8 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702223125089 7 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (f918ff0fb7f1426e99cbce10fd2fef11) <<<
```

9.2.3. Aplicació del moviment de recursos

A continuació es mostra la seqüència d'execució completa per a la prova del moviment de recursos que s'ha presentat a la secció 6.5.3 on demostrem que el sistema pot realitzar aquesta funció i distribuir els elements depenent de la càrrega de treball del *host*.

```
osgi> host_load HostPL2 0.98

20200702225751010 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostPL2) => 0.98
20200702225751018 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostPL2) => 0.98
20200702225751023 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702225751040 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (be0938fa177849bf9701dae703c13e1a)
20200702225751041 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (be0938fa177849bf9701dae703c13e1a) <<<
20200702225751112 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] REMOVING ENDPOINT 18000
20200702225751113 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] DEPLOYING CONTAINER CPL2-moved
20200702225751114 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL2-moved] DEPLOYING CONTAINER CPL2-moved IN HOST HostPL2
20200702225751115 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL2] REMOVING ENDPOINT 8080
20200702225751115 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] DELEGATE ENDPOINT 18000 TO CONTAINER CPL2-moved ENDPOINT 8080
20200702225751116 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL2-moved] EXPOSING ENDPOINT 8080
20200702225751116 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL2-moved] DEPLOYING MICROSERVICE PL2-moved
20200702225751117 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-moved] DEPLOYING MICROSERVICE PL2-moved IN CONTAINER CPL2-moved
20200702225751117 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL2] UNDEPLOYING MICROSERVICE PL2
20200702225751119 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2] UNDEPLOYING MICROSERVICE
20200702225751120 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] UNDEPLOYING CONTAINER CPL2
20200702225751120 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL2] UNDEPLOYING CONTAINER FROM HOST HostPL2
20200702225751121 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-moved] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702225751121 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-moved] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702225751122 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-moved] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702225751122 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-moved] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702225751122 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL2-moved] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL2-moved ENDPOINT pl
20200702225751122 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-moved] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702225751122 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-moved] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702225751123 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2-moved] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702225751123 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702225751176 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (be0938fa177849bf9701dae703c13e1a) <<<

osgi> cb DevPL1 high

20200702225810336 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostPL1)[Container: CDevPL1]<Microservice: DevPL1> => high
20200702225810337 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostPL1)[Container: CDevPL1]<Microservice: DevPL1> => high
20200702225810348 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule circuitbreaker-configure-rule> Asking for the next System Configuration
20200702225810354 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule circuitbreaker-configure-rule> Request Planning (3ec0ac50b9a146608d3c9cc4ccd983ee)
20200702225810359 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (3ec0ac50b9a146608d3c9cc4ccd983ee) <<<
20200702225810389 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1] UNBINDING devpl FROM HostPL1::18001
20200702225810389 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] DEPLOYING CONTAINER DevPL1-cb
20200702225810397 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [DevPL1-cb] DEPLOYING CONTAINER DevPL1-cb IN HOST HostPL2
20200702225810400 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] DELEGATE ENDPOINT 18001 TO CONTAINER DevPL1-cb ENDPOINT 8081
20200702225810400 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [DevPL1-cb] EXPOSING ENDPOINT 8081
20200702225810400 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [DevPL1-cb] DEPLOYING MICROSERVICE DevPL1-cb
20200702225810401 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL1-cb] DEPLOYING MICROSERVICE DevPL1-cb IN CONTAINER DevPL1-cb
20200702225810401 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL1-cb] EXPOSING SERVICE DevPL THROUGH INTERFACE devpl
20200702225810401 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL1-cb] DEFINING REQUIRED SERVICE DevPL THROUGH INTERFACE reqdevpl
20200702225810401 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [DevPL1-cb] DELEGATE ENDPOINT 8081 TO MICROSERVICE DevPL1-cb ENDPOINT devpl
20200702225810401 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1] PUBLIC BINDING devpl TO HOST HostPL2 ENDPOINT 18001
20200702225810402 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL1-cb] PUBLIC BINDING reqdevpl TO HOST HostPL1 ENDPOINT 18001
20200702225810402 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702225810462 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (3ec0ac50b9a146608d3c9cc4ccd983ee) <<<

osgi> cb DevPL1 low

20200702225842420278 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostPL1)[Container: CDevPL1]<Microservice: DevPL1> => low
20200702225842420278 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostPL1)[Container: CDevPL1]<Microservice: DevPL1> => low
20200702225842420281 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule circuitbreaker-configure-rule> Asking for the next System Configuration
20200702225842420281 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule circuitbreaker-configure-rule> Request Planning (02721e15a44148af93fcad266d0b7045)
20200702225842420282 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (02721e15a44148af93fcad266d0b7045) <<<
20200702225842420311 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702225842420360 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (02721e15a44148af93fcad266d0b7045) <<<
```

9.2.4. Service per Container

A continuació es mostra la seqüència d'execució completa per a la prova de l'adaptació "*Service per Container*" que s'ha presentat a la secció 6.5.4 on demostrem que el sistema pot realitzar aquesta funció i aconseguir mantindre un esquema de desplegament independentment dels canvis que es produïsquen al sistema.

```
osgi> spc
20200708172730072 1 [INFO ] ....lite.artifacts.components.Probe: (service-per-container-probe) Reporting measure: service-per-container => alert
20200708172730072 3 [DEBUG] ...ite.artifacts.components.Monitor: (Service per Container Monitor) Received measure: service-per-container => alert
20200708172730072 5 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule service-per-container-rule> Asking for the next System Configuration
20200708172730073 0 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule service-per-container-rule> Request Planning (68ad15c79e284d0ca14c277cc6c4d0dd)
20200708172730073 3 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (68ad15c79e284d0ca14c277cc6c4d0dd) <<<
20200708172730078 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] REMOVE ENDPOINT DELEGATION 8095
20200708172730078 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] REMOVE ENDPOINT DELEGATION 8094
20200708172730078 4 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DEPLOYING CONTAINER ThingsTest-moved
20200708172730078 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [ThingsTest-moved] DEPLOYING CONTAINER ThingsTest-moved IN HOST HostERP
20200708172730078 5 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DEPLOYING CONTAINER ThingsProva-moved
20200708172730078 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [ThingsProva-moved] DEPLOYING CONTAINER ThingsProva-moved IN HOST HostERP
20200708172730078 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] REMOVING ENDPOINT 8095
20200708172730078 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] REMOVING ENDPOINT 8094
20200708172730078 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [ThingsTest-moved] EXPOSING ENDPOINT 8095
20200708172730078 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [ThingsProva-moved] EXPOSING ENDPOINT 8094
20200708172730078 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] UNDEPLOYING MICROSERVICE ThingsTest
20200708172730078 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [ThingsTest] UNDEPLOYING MICROSERVICE
20200708172730079 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CThings] UNDEPLOYING MICROSERVICE ThingsProva
20200708172730079 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [ThingsProva] UNDEPLOYING MICROSERVICE
20200708172730079 2 [DEBUG] ...pek.lite.modules.impl.LoopModule: Adaptation aborted. Executing exception
20200708172730079 9 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (68ad15c79e284d0ca14c277cc6c4d0dd) <<<
```

9.2.5. Adaptacions autònomes

A continuació es mostra la seqüència d'execució completa per a la prova de l'adaptació autònoma que s'ha presentat a la secció 6.5.5 on demostrem que el sistema pot realitzar aquesta funció i auto-gestionar-se utilitzant els patrons que hem decidit incloure i presentar en seccions anteriors.

```
osgi> simulator start

20200702230317091 9 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: PL3 => very-high
20200702230317092 1 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: PL3 => very-high
20200702230317092 3 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule LOAD-BALANCER> Asking for the next System Configuration
20200702230317093 7 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule LOAD-BALANCER> Request Planning (660fe783b71940a2b0ad450a4880847a)
20200702230317094 0 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: DevTLPL1 => low
20200702230317094 1 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (660fe783b71940a2b0ad450a4880847a) <<<
20200702230317094 1 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: DevTLPL1 => low
20200702230317094 2 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: TLC1 => high
20200702230317094 2 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: TLC1 => high
20200702230317094 3 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule LOAD-BALANCER> Asking for the next System Configuration
20200702230317095 2 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule LOAD-BALANCER> Request Planning (acdaf853228640798a29df02adb9274b)
20200702230317095 4 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: TLC3 => low
20200702230317095 4 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: TLC3 => low
20200702230317095 5 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: things => low
20200702230317095 5 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: things => low
20200702230317095 8 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostTL1) => 2.2666667
20200702230317095 9 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostTL1) => 2.2666667
20200702230317096 1 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230317097 0 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (f0e941f7cc244499a5eee08f0cdf4abf)
20200702230317097 1 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostTL2) => 1.6666666
20200702230317097 1 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostTL2) => 1.6666666
20200702230317097 5 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230317098 2 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (a8c8c472550b4ec1b92e0963d84adfae)
20200702230317098 2 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostTL3) => 1.117647
20200702230317098 3 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostTL3) => 1.117647
20200702230317098 4 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230317099 5 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (e6cba86c2e6f497791918bd799f24b3d)
20200702230317099 6 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostTL4) => 2.25
20200702230317099 6 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostTL4) => 2.25
20200702230317099 7 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230318000 7 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (14e8be7127184dcfb2ef69b51488eba1)
20200702230318000 9 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostERP) => 0.9791667
20200702230318000 9 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostERP) => 0.9791667
20200702230318001 1 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230318001 7 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (9ade873b1a8f4b0dbd79820afd420291)
```

```
20200702230318006 5 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DEPLOYING CONTAINER PL3-lb
20200702230318006 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-lb] DEPLOYING CONTAINER PL3-lb IN HOST HostPL1
20200702230318007 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL3] DEPLOYING CONTAINER PL3-1
20200702230318007 3 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-1] DEPLOYING CONTAINER PL3-1 IN HOST HostPL3
20200702230318007 4 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW2] DEPLOYING CONTAINER PL3-2
20200702230318007 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-2] DEPLOYING CONTAINER PL3-2 IN HOST HostREW2
20200702230318007 5 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL4] DEPLOYING CONTAINER PL3-3
20200702230318007 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-3] DEPLOYING CONTAINER PL3-3 IN HOST HostPL4
20200702230318007 6 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] DEPLOYING CONTAINER PL3-4
20200702230318007 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-4] DEPLOYING CONTAINER PL3-4 IN HOST HostPL2
20200702230318007 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW1] DEPLOYING CONTAINER PL3-5
20200702230318007 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-5] DEPLOYING CONTAINER PL3-5 IN HOST HostREW1
20200702230318007 9 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostERP] DEPLOYING CONTAINER PL3-6
20200702230318007 9 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-6] DEPLOYING CONTAINER PL3-6 IN HOST HostERP
20200702230318008 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DELEGATE ENDPOINT 18000 TO CONTAINER PL3-lb ENDPOINT 8080
20200702230318008 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL3] DELEGATE ENDPOINT 18000 TO CONTAINER PL3-1 ENDPOINT 8080
20200702230318008 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-lb] EXPOSING ENDPOINT 8080
20200702230318008 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-1] EXPOSING ENDPOINT 8080
20200702230318008 1 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-2] EXPOSING ENDPOINT 8080
20200702230318008 1 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-3] EXPOSING ENDPOINT 8080
20200702230318008 1 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-4] EXPOSING ENDPOINT 8080
20200702230318008 1 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-5] EXPOSING ENDPOINT 8080
20200702230318008 1 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-6] EXPOSING ENDPOINT 8080
20200702230318008 2 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-1] DEPLOYING MICROSERVICE PL3-1
20200702230318008 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1] DEPLOYING MICROSERVICE PL3-1 IN CONTAINER PL3-1
20200702230318008 3 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-2] DEPLOYING MICROSERVICE PL3-2
20200702230318008 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-2] DEPLOYING MICROSERVICE PL3-2 IN CONTAINER PL3-2
20200702230318008 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-3] DEPLOYING MICROSERVICE PL3-3
20200702230318008 5 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3] DEPLOYING MICROSERVICE PL3-3 IN CONTAINER PL3-3
20200702230318008 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-4] DEPLOYING MICROSERVICE PL3-4
20200702230318008 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-4] DEPLOYING MICROSERVICE PL3-4 IN CONTAINER PL3-4
20200702230318008 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-5] DEPLOYING MICROSERVICE PL3-5
20200702230318008 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-5] DEPLOYING MICROSERVICE PL3-5 IN CONTAINER PL3-5
20200702230318008 9 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-6] DEPLOYING MICROSERVICE PL3-6
20200702230318008 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-6] DEPLOYING MICROSERVICE PL3-6 IN CONTAINER PL3-6
20200702230318009 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-lb] DEPLOYING MICROSERVICE PL3-lb
20200702230318009 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-lb] DEPLOYING MICROSERVICE PL3-lb IN CONTAINER PL3-lb
20200702230318009 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702230318009 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-2] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702230318009 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702230318009 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-4] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702230318009 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-5] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702230318009 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-6] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702230318009 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-lb] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702230318009 5 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702230318009 5 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702230318009 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702230318009 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-2] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702230318009 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-2] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702230318009 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-2] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702230318009 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702230318009 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702230318009 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702230318009 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-4] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702230318009 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-4] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702230318009 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-4] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702230318009 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-5] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702230318009 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-5] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702230318009 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-5] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702230318009 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-6] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702230318009 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-6] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702230318009 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-6] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702230318009 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-lb] DEFINING REQUIRED SERVICE PL THROUGH INTERFACE reqpl
20200702230318009 9 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-lb] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL3-lb ENDPOINT pl
20200702230318010 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-1] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL3-1 ENDPOINT pl
20200702230318010 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-2] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL3-2 ENDPOINT pl
20200702230318010 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-3] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL3-3 ENDPOINT pl
20200702230318010 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-4] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL3-4 ENDPOINT pl
20200702230318010 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-5] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL3-5 ENDPOINT pl
20200702230318010 1 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-6] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL3-6 ENDPOINT pl
20200702230318010 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702230318010 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702230318010 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702230318010 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-2] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702230318010 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-2] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702230318010 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-2] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702230318010 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702230318010 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702230318010 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702230318010 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-4] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702230318010 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-4] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702230318010 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-4] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702230318010 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-5] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702230318010 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-5] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702230318010 5 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-5] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702230318010 5 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-6] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702230318010 5 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-6] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702230318010 5 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-6] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702230318010 5 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702230318022 8 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (660fe783b71940a2b0ad450a4880847a) <<<
20200702230318022 9 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (acdaf853228640798a29df02adb9274b) <<<
20200702230318028 7 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] EXPOSING EDNPOINT 18020
20200702230318028 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL3] EXPOSING EDNPOINT 18020
20200702230318028 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW2] EXPOSING EDNPOINT 18020
20200702230318028 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL4] EXPOSING EDNPOINT 18020
20200702230318028 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DEPLOYING CONTAINER TLC1-lb
20200702230318028 9 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-lb] DEPLOYING CONTAINER TLC1-lb IN HOST HostPL1
20200702230318028 9 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL3] DEPLOYING CONTAINER TLC1-1
20200702230318029 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-1] DEPLOYING CONTAINER TLC1-1 IN HOST HostPL3
20200702230318029 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW2] DEPLOYING CONTAINER TLC1-2
20200702230318029 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-2] DEPLOYING CONTAINER TLC1-2 IN HOST HostREW2
20200702230318029 1 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL4] DEPLOYING CONTAINER TLC1-3
20200702230318029 1 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-3] DEPLOYING CONTAINER TLC1-3 IN HOST HostPL4
20200702230318029 2 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DELEGATE ENDPOINT 18020 TO CONTAINER TLC1-lb ENDPOINT 8084
20200702230318029 2 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL3] DELEGATE ENDPOINT 18020 TO CONTAINER TLC1-1 ENDPOINT 8084
20200702230318029 2 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW2] DELEGATE ENDPOINT 18020 TO CONTAINER TLC1-2 ENDPOINT 8084
20200702230318029 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL4] DELEGATE ENDPOINT 18020 TO CONTAINER TLC1-3 ENDPOINT 8084
20200702230318029 3 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-lb] EXPOSING ENDPOINT 8084
```

```
20200702230318029 3 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-1] EXPOSING ENDPOINT 8084
20200702230318029 3 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-2] EXPOSING ENDPOINT 8084
20200702230318029 3 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-3] EXPOSING ENDPOINT 8084
20200702230318029 3 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-1] DEPLOYING MICROSERVICE TLC1-1
20200702230318029 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-1] DEPLOYING MICROSERVICE TLC1-1 IN CONTAINER TLC1-1
20200702230318029 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-2] DEPLOYING MICROSERVICE TLC1-2
20200702230318029 5 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2] DEPLOYING MICROSERVICE TLC1-2 IN CONTAINER TLC1-2
20200702230318029 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-3] DEPLOYING MICROSERVICE TLC1-3
20200702230318029 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-3] DEPLOYING MICROSERVICE TLC1-3 IN CONTAINER TLC1-3
20200702230318029 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-lb] DEPLOYING MICROSERVICE TLC1-lb
20200702230318029 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-lb] DEPLOYING MICROSERVICE TLC1-lb IN CONTAINER TLC1-lb
20200702230318029 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-1] EXPOSING SERVICE TLC THROUGH INTERFACE tlcpl
20200702230318029 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2] EXPOSING SERVICE TLC THROUGH INTERFACE tlcpl
20200702230318029 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-3] EXPOSING SERVICE TLC THROUGH INTERFACE tlcpl
20200702230318029 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-lb] EXPOSING SERVICE TLC THROUGH INTERFACE tlcpl
20200702230318029 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-1] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702230318029 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-1] DEFINING REQUIRED SERVICE DevPL THROUGH INTERFACE devpl
20200702230318029 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-1] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702230318029 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-1] DEFINING REQUIRED SERVICE DevTL THROUGH INTERFACE devtl
20200702230318029 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702230318029 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2] DEFINING REQUIRED SERVICE DevPL THROUGH INTERFACE devpl
20200702230318029 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702230318029 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2] DEFINING REQUIRED SERVICE DevTL THROUGH INTERFACE devtl
20200702230318029 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-3] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702230318029 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-3] DEFINING REQUIRED SERVICE DevPL THROUGH INTERFACE devpl
20200702230318029 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-3] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702230318029 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-3] DEFINING REQUIRED SERVICE DevTL THROUGH INTERFACE devtl
20200702230318030 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-lb] DEFINING REQUIRED SERVICE TLC THROUGH INTERFACE reqtlcpl
20200702230318030 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-lb] DELEGATE ENDPOINT 8084 TO MICROSERVICE TLC1-lb ENDPOINT tlcpl
20200702230318030 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-1] DELEGATE ENDPOINT 8084 TO MICROSERVICE TLC1-1 ENDPOINT tlcpl
20200702230318030 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-2] DELEGATE ENDPOINT 8084 TO MICROSERVICE TLC1-2 ENDPOINT tlcpl
20200702230318030 1 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-3] DELEGATE ENDPOINT 8084 TO MICROSERVICE TLC1-3 ENDPOINT tlcpl
20200702230318030 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-1] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702230318030 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-1] PUBLIC BINDING devpl TO HOST HostPL1 ENDPOINT 18001
20200702230318030 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-1] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702230318030 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-1] PUBLIC BINDING devtl TO HOST HostTL1 ENDPOINT 18021
20200702230318030 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702230318030 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2] PUBLIC BINDING devpl TO HOST HostPL1 ENDPOINT 18001
20200702230318030 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702230318030 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2] PUBLIC BINDING devtl TO HOST HostTL1 ENDPOINT 18021
20200702230318030 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-3] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702230318030 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-3] PUBLIC BINDING devpl TO HOST HostPL1 ENDPOINT 18001
20200702230318030 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-3] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702230318030 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-3] PUBLIC BINDING devtl TO HOST HostTL1 ENDPOINT 18021
20200702230318030 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-lb] PUBLIC BINDING reqtlcpl TO HOST HostPL3 ENDPOINT 18020
20200702230318030 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-lb] PUBLIC BINDING reqtlcpl TO HOST HostREW2 ENDPOINT 18020
20200702230318030 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-lb] PUBLIC BINDING reqtlcpl TO HOST HostPL4 ENDPOINT 18020
20200702230318030 4 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702230318036 6 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (acdaf853228640798a29df02adb9274b) <<<
20200702230318036 6 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (f0e941f7cc244499a5eee08f0cdf4abf) <<<
20200702230318044 2 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL1] REMOVING ENDPOINT 18020
20200702230318044 2 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DEPLOYING CONTAINER CTLC1-moved
20200702230318044 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC1-moved] DEPLOYING CONTAINER CTLC1-moved IN HOST HostPL1
20200702230318044 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC1] REMOVING ENDPOINT 8084
20200702230318044 5 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DELEGATE ENDPOINT 18020 TO CONTAINER CTLC1-moved ENDPOINT 8084
20200702230318044 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC1-moved] EXPOSING ENDPOINT 8084
20200702230318044 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC1-moved] DEPLOYING MICROSERVICE TLC1-moved
20200702230318044 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-moved] DEPLOYING MICROSERVICE TLC1-moved IN CONTAINER CTLC1-moved
20200702230318044 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC1] UNDEPLOYING MICROSERVICE TLC1
20200702230318044 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1] UNDEPLOYING MICROSERVICE
20200702230318044 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL1] UNDEPLOYING CONTAINER CTLC1
20200702230318044 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC1] UNDEPLOYING CONTAINER FROM HOST HostTL1
20200702230318044 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-moved] EXPOSING SERVICE TLC THROUGH INTERFACE tlcpl
20200702230318044 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-moved] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702230318044 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-moved] DEFINING REQUIRED SERVICE DevPL THROUGH INTERFACE devpl
20200702230318044 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-moved] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702230318044 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-moved] DEFINING REQUIRED SERVICE DevTL THROUGH INTERFACE devtl
20200702230318044 9 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC1-moved] DELEGATE ENDPOINT 8084 TO MICROSERVICE TLC1-moved ENDPOINT tlcpl
20200702230318044 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-moved] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702230318044 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-moved] PUBLIC BINDING devpl TO HOST HostPL1 ENDPOINT 18001
20200702230318045 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-moved] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702230318045 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-moved] PUBLIC BINDING devtl TO HOST HostTL1 ENDPOINT 18021
20200702230318045 0 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702230318048 4 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (f0e941f7cc244499a5eee08f0cdf4abf) <<<
20200702230318048 5 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (a8c8c472550b4ec1b92e0963d84adfae) <<<
20200702230318051 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2] UNBINDING devtl FROM HostTL2::18021
20200702230318051 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] EXPOSING EDNPOINT 18021
20200702230318051 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] REMOVING ENDPOINT 18021
20200702230318051 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DEPLOYING CONTAINER CDevTLPL2-moved
20200702230318051 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL2-moved] DEPLOYING CONTAINER CDevTLPL2-moved IN HOST HostPL1
20200702230318051 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL2] REMOVING ENDPOINT 8085
20200702230318051 4 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DELEGATE ENDPOINT 18021 TO CONTAINER CDevTLPL2-moved ENDPOINT 8085
20200702230318051 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL2-moved] EXPOSING ENDPOINT 8085
20200702230318051 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL2-moved] DEPLOYING MICROSERVICE DevTLPL2-moved
20200702230318051 5 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2-moved] DEPLOYING MICROSERVICE DevTLPL2-moved IN CONTAINER CDevTLPL2-moved
20200702230318051 6 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL2] UNDEPLOYING MICROSERVICE DevTLPL2
20200702230318051 6 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2] UNDEPLOYING MICROSERVICE
20200702230318051 7 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] UNDEPLOYING CONTAINER CDevTLPL2
20200702230318051 7 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL2] UNDEPLOYING CONTAINER FROM HOST HostTL2
20200702230318051 7 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2-moved] EXPOSING SERVICE DevTL THROUGH INTERFACE devtl
20200702230318051 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2-moved] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702230318051 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2-moved] DEFINING REQUIRED SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702230318051 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL2-moved] DELEGATE ENDPOINT 8085 TO MICROSERVICE DevTLPL2-moved ENDPOINT devtl
20200702230318051 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2-moved] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702230318051 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2-moved] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702230318051 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2] PUBLIC BINDING devtl TO HOST HostPL1 ENDPOINT 18021
20200702230318051 9 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702230318055 7 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (a8c8c472550b4ec1b92e0963d84adfae) <<<
20200702230318055 9 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (e6cba86c2e6f497791918bd799f24b3d) <<<
20200702230318058 9 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] REMOVING ENDPOINT 18020
20200702230318058 9 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DEPLOYING CONTAINER CTLC3-moved
20200702230318058 9 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC3-moved] DEPLOYING CONTAINER CTLC3-moved IN HOST HostPL1
20200702230318058 9 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC3] REMOVING ENDPOINT 8084
20200702230318059 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DELEGATE ENDPOINT 18020 TO CONTAINER CTLC3-moved ENDPOINT 8084
20200702230318059 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC3-moved] EXPOSING ENDPOINT 8084
20200702230318059 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC3-moved] DEPLOYING MICROSERVICE TLC3-moved
20200702230318059 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3-moved] DEPLOYING MICROSERVICE TLC3-moved IN CONTAINER CTLC3-moved
20200702230318059 1 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC3] UNDEPLOYING MICROSERVICE TLC3
```

```
20200702230318059l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3] UNDEPLOYING MICROSERVICE
20200702230318059l [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] UNDEPLOYING CONTAINER CTLC3
20200702230318059l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC3] UNDEPLOYING CONTAINER FROM HOST HostTL3
20200702230318059l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3-moved] EXPOSING SERVICE TLC THROUGH INTERFACE tlcpl
20200702230318059l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3-moved] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702230318059l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3-moved] DEFINING REQUIRED SERVICE DevPL THROUGH INTERFACE devpl
20200702230318059l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3-moved] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702230318059l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3-moved] DEFINING REQUIRED SERVICE DevTL THROUGH INTERFACE devtl
20200702230318059l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC3-moved] DELEGATE ENDPOINT 8084 TO MICROSERVICE TLC3-moved ENDPOINT tlcpl
20200702230318059l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3-moved] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702230318059l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3-moved] PUBLIC BINDING devpl TO HOST HostPL3 ENDPOINT 18001
20200702230318059l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3-moved] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702230318059l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC3-moved] PUBLIC BINDING devtl TO HOST HostTL3 ENDPOINT 18021
20200702230318059l [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702230318064l [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (e6cba86c2e6f497791918bd799f24b3d) <<<
20200702230318064l [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (14e8be7127184dcfb2ef69b51488eba1) <<<
20200702230318066l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4] UNBINDING devtl FROM HostTL4::18021
20200702230318066l [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL4] REMOVING ENDPOINT 18021
20200702230318066l [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DEPLOYING CONTAINER CDevTLPL4-moved
20200702230318066l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL4-moved] DEPLOYING CONTAINER CDevTLPL4-moved IN HOST HostPL1
20200702230318066l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL4] REMOVING ENDPOINT 8085
20200702230318066l [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] DELEGATE ENDPOINT 18021 TO CONTAINER CDevTLPL4-moved ENDPOINT 8085
20200702230318066l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL4-moved] EXPOSING ENDPOINT 8085
20200702230318066l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL4-moved] DEPLOYING MICROSERVICE DevTLPL4-moved
20200702230318066l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4-moved] DEPLOYING MICROSERVICE DevTLPL4-moved IN CONTAINER CDevTLPL4-moved
20200702230318067l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL4] UNDEPLOYING MICROSERVICE DevTLPL4
20200702230318067l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4] UNDEPLOYING MICROSERVICE
20200702230318067l [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL4] UNDEPLOYING CONTAINER CDevTLPL4
20200702230318067l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL4] UNDEPLOYING CONTAINER FROM HOST HostTL4
20200702230318067l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4-moved] EXPOSING SERVICE DevTL THROUGH INTERFACE devtl
20200702230318067l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4-moved] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702230318067l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4-moved] DEFINING REQUIRED SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702230318067l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL4-moved] DELEGATE ENDPOINT 8085 TO MICROSERVICE DevTLPL4-moved ENDPOINT devtl
20200702230318067l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4-moved] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702230318067l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4-moved] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702230318067l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4] PUBLIC BINDING devtl TO HOST HostPL1 ENDPOINT 18021
20200702230318067l [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702230318070l [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (14e8be7127184dcfb2ef69b51488eba1) <<<
20200702230318070l [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (9ade873b1a8f4b0dbd79820afd420291) <<<
20200702230318072l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1] UNBINDING plwo FROM HostERP::18030
20200702230318072l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2] UNBINDING plwo FROM HostERP::18030
20200702230318072l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3] UNBINDING plwo FROM HostERP::18030
20200702230318072l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL4] UNBINDING plwo FROM HostERP::18030
20200702230318072l [DEBUG] ...pek.lite.modules.impl.LoopModule: Adaptation aborted. Executing exception
20200702230318072l [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (9ade873b1a8f4b0dbd79820afd420291) <<<
20200702230327091l [INFO ] ....lite.artifacts.components.Probe: Reporting measure: PL2 => low
20200702230327091l [DEBUG] ...ite.artifacts.components.Monitor: Received measure: PL2 => low
20200702230327092l [INFO ] ....lite.artifacts.components.Probe: Reporting measure: TLC4 => low
20200702230327092l [DEBUG] ...ite.artifacts.components.Monitor: Received measure: TLC4 => low
20200702230327092l [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostPL1) => 1.4320987
20200702230327092l [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostPL1) => 1.4320987
20200702230327092l [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230327093l [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (8f45bac96d07497cb62eee5a59d76510)
20200702230327093l [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (8f45bac96d07497cb62eee5a59d76510) <<<
20200702230327093l [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostPL2) => 1.0763888
20200702230327093l [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostPL2) => 1.0763888
20200702230327093l [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230327093l [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (6d37ba267a994b76977637c39934c3db)
20200702230327093l [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostPL3) => 0.974359
20200702230327093l [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostPL3) => 0.974359
20200702230327093l [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230327094l [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (2f21eb84c3f944ebbc0ad3cae33ea285)
20200702230327094l [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostPL4) => 1.6041666
20200702230327094l [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostPL4) => 1.6041666
20200702230327094l [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230327094l [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (d28c18e1170e4d73ab44740aef700f3b)
20200702230327094l [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostREW1) => 1.5384616
20200702230327094l [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostREW1) => 1.5384616
20200702230327094l [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230327095l [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (824eb0c38cb1482abb61ffe16db335db)
20200702230327095l [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostREW2) => 1.2322581
20200702230327095l [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostREW2) => 1.2322581
20200702230327095l [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230327095l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2] UNBINDING devtl FROM HostPL1::18021
20200702230327095l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4] UNBINDING devtl FROM HostPL1::18021
20200702230327095l [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] REMOVING ENDPOINT 18021
20200702230327095l [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DEPLOYING CONTAINER CDevTLPL4-moved-moved
20200702230327095l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL4-moved-moved] DEPLOYING CONTAINER CDevTLPL4-moved-moved IN HOST HostTL3
20200702230327095l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL4-moved] REMOVING ENDPOINT 8085
20200702230327095l [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DELEGATE ENDPOINT 18021 TO CONTAINER CDevTLPL4-moved-moved ENDPOINT 8085
20200702230327095l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL4-moved-moved] EXPOSING ENDPOINT 8085
20200702230327095l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL4-moved-moved] DEPLOYING MICROSERVICE DevTLPL4-moved-moved
20200702230327096l [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (1b0f37fc4dd94cc8a42045ece01a664b)
20200702230327096l [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostTL1) => 1.1666666
20200702230327096l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4-moved-moved] DEPLOYING MICROSERVICE DevTLPL4-moved-moved IN CONTAINER CDevTLPL4-
moved-moved
20200702230327096l [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostTL1) => 1.1666666
20200702230327096l [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230327096l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL4-moved] UNDEPLOYING MICROSERVICE DevTLPL4-moved
20200702230327096l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4-moved] UNDEPLOYING MICROSERVICE
20200702230327096l [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] UNDEPLOYING CONTAINER CDevTLPL4-moved
20200702230327096l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL4-moved] UNDEPLOYING CONTAINER FROM HOST HostPL1
20200702230327096l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4-moved-moved] EXPOSING SERVICE DevTL THROUGH INTERFACE devtl
20200702230327096l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4-moved-moved] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702230327096l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4-moved-moved] DEFINING REQUIRED SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702230327096l [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL4-moved-moved] DELEGATE ENDPOINT 8085 TO MICROSERVICE DevTLPL4-moved-moved
ENDPOINT devtl
20200702230327096l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4-moved-moved] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702230327096l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4-moved-moved] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702230327096l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4-moved-moved] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702230327096l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL4-moved-moved] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702230327096l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2] PUBLIC BINDING devtl TO HOST HostTL3 ENDPOINT 18021
20200702230327096l [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4] PUBLIC BINDING devtl TO HOST HostTL3 ENDPOINT 18021
20200702230327096l [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702230327097l [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (b201ba0b6ac74ab5b0d172fa8455eb24)
20200702230327097l [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostTL2) => 1.037037
20200702230327097l [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostTL2) => 1.037037
20200702230327097l [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
```


```
20200702230270976 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (f0cd486674b24b5cadf9a8f486bcb72f)
20200702230270977 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostTL4) => 1.0
20200702230270977 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostTL4) => 1.0
20200702230270979 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230270988 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (c1643893d7e94918870f410537c42d11)
20200702230270989 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostERP) => 1.0833334
20200702230270989 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostERP) => 1.0833334
20200702230270990 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230270994 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (09aae63dbf544d3393213037b846cf6d)
20200702230280005 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (8f45bac96d07497cb62eee5a59d76510) <<<
20200702230280006 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (6d37ba267a994b76977637c39934c3db) <<<
20200702230280045 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2] UNBINDING devpl FROM HostPL2::18001
20200702230280045 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] EXPOSING EDNPOINT 18001
20200702230280045 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] REMOVING ENDPOINT 18001
20200702230280045 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DEPLOYING CONTAINER CDevPL2-moved
20200702230280046 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL2-moved] DEPLOYING CONTAINER CDevPL2-moved IN HOST HostTL3
20200702230280047 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL2] REMOVING ENDPOINT 8081
20200702230280047 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DELEGATE ENDPOINT 18001 TO CONTAINER CDevPL2-moved ENDPOINT 8081
20200702230280047 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL2-moved] EXPOSING ENDPOINT 8081
20200702230280047 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL2-moved] DEPLOYING MICROSERVICE DevPL2-moved
20200702230280048 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL2-moved] DEPLOYING MICROSERVICE DevPL2-moved IN CONTAINER CDevPL2-moved
20200702230280048 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL2] UNDEPLOYING MICROSERVICE DevPL2
20200702230280049 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL2] UNDEPLOYING MICROSERVICE
20200702230280049 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] UNDEPLOYING CONTAINER CDevPL2
20200702230280050 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL2] UNDEPLOYING CONTAINER FROM HOST HostPL2
20200702230280050 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL2-moved] EXPOSING SERVICE DevPL THROUGH INTERFACE devpl
20200702230280051 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL2-moved] DEFINING REQUIRED SERVICE Locations THROUGH INTERFACE loc
20200702230280051 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL2-moved] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702230280051 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL2-moved] DEFINING REQUIRED SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702230280052 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevPL2-moved] DELEGATE ENDPOINT 8081 TO MICROSERVICE DevPL2-moved ENDPOINT devpl
20200702230280052 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL2-moved] PUBLIC BINDING loc TO HOST HostERP ENDPOINT 18040
20200702230280052 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL2-moved] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702230280052 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevPL2-moved] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702230280052 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2] PUBLIC BINDING devpl TO HOST HostTL3 ENDPOINT 18001
20200702230280053 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702230280093 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (6d37ba267a994b76977637c39934c3db) <<<
20200702230280093 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (2f21eb84c3f944ebbc0ad3cae33ea285) <<<
20200702230280122 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] EXPOSING EDNPOINT 18000
20200702230280122 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL3] REMOVING ENDPOINT 18000
20200702230280123 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DEPLOYING CONTAINER PL3-1-moved
20200702230280123 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-1-moved] DEPLOYING CONTAINER PL3-1-moved IN HOST HostTL3
20200702230280125 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-1] REMOVING ENDPOINT 8080
20200702230280125 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DELEGATE ENDPOINT 18000 TO CONTAINER PL3-1-moved ENDPOINT 8080
20200702230280126 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-1-moved] EXPOSING ENDPOINT 8080
20200702230280126 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-1-moved] DEPLOYING MICROSERVICE PL3-1-moved
20200702230280134 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1-moved] DEPLOYING MICROSERVICE PL3-1-moved IN CONTAINER PL3-1-moved
20200702230280134 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-1] UNDEPLOYING MICROSERVICE PL3-1
20200702230280135 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1] UNDEPLOYING MICROSERVICE
20200702230280136 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL3] UNDEPLOYING CONTAINER PL3-1
20200702230280136 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-1] UNDEPLOYING CONTAINER FROM HOST HostPL3
20200702230280137 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1-moved] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702230280137 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1-moved] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702230280137 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1-moved] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702230280137 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1-moved] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702230280137 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-1-moved] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL3-1-moved ENDPOINT pl
20200702230280137 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1-moved] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702230280138 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1-moved] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702230280138 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1-moved] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702230280138 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1-moved] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702230280138 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1-moved] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702230280138 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-1-moved] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702230280138 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702230280198 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (2f21eb84c3f944ebbc0ad3cae33ea285) <<<
20200702230280199 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (d28c18e1170e4d73ab44740aef700f3b) <<<
20200702230280222 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DEPLOYING CONTAINER PL3-3-moved
20200702230280223 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-3-moved] DEPLOYING CONTAINER PL3-3-moved IN HOST HostTL3
20200702230280223 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-3] REMOVING ENDPOINT 8080
20200702230280223 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-3-moved] EXPOSING ENDPOINT 8080
20200702230280223 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-3-moved] DEPLOYING MICROSERVICE PL3-3-moved
20200702230280224 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3-moved] DEPLOYING MICROSERVICE PL3-3-moved IN CONTAINER PL3-3-moved
20200702230280224 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-3] UNDEPLOYING MICROSERVICE PL3-3
20200702230280225 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3] UNDEPLOYING MICROSERVICE
20200702230280225 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL4] UNDEPLOYING CONTAINER PL3-3
20200702230280225 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-3] UNDEPLOYING CONTAINER FROM HOST HostPL4
20200702230280226 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3-moved] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702230280226 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3-moved] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702230280226 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3-moved] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702230280226 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3-moved] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702230280226 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-3-moved] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL3-3-moved ENDPOINT pl
20200702230280226 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3-moved] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702230280226 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3-moved] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702230280227 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3-moved] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702230280227 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3-moved] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702230280227 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3-moved] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702230280227 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3-3-moved] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702230280227 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702230280274 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (d28c18e1170e4d73ab44740aef700f3b) <<<
20200702230280275 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (824eb0c38cb1482abb61ffe16db335db) <<<
20200702230280294 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] EXPOSING EDNPOINT 18011
20200702230280294 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW1] REMOVING ENDPOINT 18011
20200702230280295 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DEPLOYING CONTAINER CDevREW1-moved
20200702230280295 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevREW1-moved] DEPLOYING CONTAINER CDevREW1-moved IN HOST HostTL3
20200702230280295 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevREW1] REMOVING ENDPOINT 8083
20200702230280295 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DELEGATE ENDPOINT 18011 TO CONTAINER CDevREW1-moved ENDPOINT 8083
20200702230280296 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevREW1-moved] EXPOSING ENDPOINT 8083
20200702230280296 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevREW1-moved] DEPLOYING MICROSERVICE DevREW1-moved
20200702230280296 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW1-moved] DEPLOYING MICROSERVICE DevREW1-moved IN CONTAINER CDevREW1-moved
20200702230280297 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevREW1] UNDEPLOYING MICROSERVICE DevREW1
20200702230280297 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW1] UNDEPLOYING MICROSERVICE
20200702230280297 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW1] UNDEPLOYING CONTAINER CDevREW1
20200702230280297 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevREW1] UNDEPLOYING CONTAINER FROM HOST HostREW1
20200702230280298 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW1-moved] EXPOSING SERVICE DevREW THROUGH INTERFACE devrew
20200702230280298 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW1-moved] DEFINING REQUIRED SERVICE Locations THROUGH INTERFACE loc
20200702230280299 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW1-moved] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702230280299 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW1-moved] DEFINING REQUIRED SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702230280299 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevREW1-moved] DELEGATE ENDPOINT 8083 TO MICROSERVICE DevREW1-moved ENDPOINT devrew
20200702230280299 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW1-moved] PUBLIC BINDING loc TO HOST HostERP ENDPOINT 18040
20200702230280299 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW1-moved] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
```

```
20200702230328029 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevREW1-moved] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702230328029 9 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702230328032 7 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (824eb0c38cb1482abb61ffe16db335db) <<<
20200702230328032 8 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (1b0f37fc4dd94cc8a42045ece01a664b) <<<
20200702230328035 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-lb] UNBINDING reqtlcpl FROM HostREW2::18020
20200702230328035 9 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] EXPOSING EDNPOINT 18020
20200702230328035 9 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW2] REMOVING ENDPOINT 18020
20200702230328036 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DEPLOYING CONTAINER TLC1-2-moved
20200702230328036 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-2-moved] DEPLOYING CONTAINER TLC1-2-moved IN HOST HostTL3
20200702230328036 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-2] REMOVING ENDPOINT 8084
20200702230328036 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DELEGATE ENDPOINT 18020 TO CONTAINER TLC1-2-moved ENDPOINT 8084
20200702230328036 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-2-moved] EXPOSING ENDPOINT 8084
20200702230328036 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-2-moved] DEPLOYING MICROSERVICE TLC1-2-moved
20200702230328036 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2-moved] DEPLOYING MICROSERVICE TLC1-2-moved IN CONTAINER TLC1-2-moved
20200702230328036 1 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-2] UNDEPLOYING MICROSERVICE TLC1-2
20200702230328036 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2] UNDEPLOYING MICROSERVICE
20200702230328036 2 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW2] UNDEPLOYING CONTAINER TLC1-2
20200702230328036 2 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-2] UNDEPLOYING CONTAINER FROM HOST HostREW2
20200702230328036 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2-moved] EXPOSING SERVICE TLC THROUGH INTERFACE tlcpl
20200702230328036 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2-moved] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702230328036 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2-moved] DEFINING REQUIRED SERVICE DevPL THROUGH INTERFACE devpl
20200702230328036 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2-moved] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702230328036 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2-moved] DEFINING REQUIRED SERVICE DevTL THROUGH INTERFACE devtl
20200702230328036 2 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [TLC1-2-moved] DELEGATE ENDPOINT 8084 TO MICROSERVICE TLC1-2-moved ENDPOINT tlcpl
20200702230328036 2 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2-moved] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702230328036 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2-moved] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702230328036 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2-moved] PUBLIC BINDING devpl TO HOST HostPL1 ENDPOINT 18001
20200702230328036 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2-moved] PUBLIC BINDING devpl TO HOST HostPL1 ENDPOINT 18001
20200702230328036 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2-moved] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702230328036 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2-moved] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702230328036 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2-moved] PUBLIC BINDING devtl TO HOST HostTL1 ENDPOINT 18021
20200702230328036 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-2-moved] PUBLIC BINDING devtl TO HOST HostTL1 ENDPOINT 18021
20200702230328036 3 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC1-lb] PUBLIC BINDING reqtlcpl TO HOST HostTL3 ENDPOINT 18020
20200702230328036 3 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702230328041 1 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (1b0f37fc4dd94cc8a42045ece01a664b) <<<
20200702230328042 1 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (b201ba0b6ac74ab5b0d172fa8455eb24) <<<
20200702230328044 3 [DEBUG] ...pek.lite.modules.impl.LoopModule: Adaptation aborted. Executing exception
20200702230328044 4 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (b201ba0b6ac74ab5b0d172fa8455eb24) <<<
20200702230328044 4 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (f0cd486674b24b5cadf9a8f486bcb72f) <<<
20200702230328046 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] REMOVING ENDPOINT 18020
20200702230328046 3 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DEPLOYING CONTAINER CTLC2-moved
20200702230328046 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC2-moved] DEPLOYING CONTAINER CTLC2-moved IN HOST HostTL3
20200702230328046 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC2] REMOVING ENDPOINT 8084
20200702230328046 4 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DELEGATE ENDPOINT 18020 TO CONTAINER CTLC2-moved ENDPOINT 8084
20200702230328046 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC2-moved] EXPOSING ENDPOINT 8084
20200702230328046 4 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC2-moved] DEPLOYING MICROSERVICE TLC2-moved
20200702230328046 4 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2-moved] DEPLOYING MICROSERVICE TLC2-moved IN CONTAINER CTLC2-moved
20200702230328046 5 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC2] UNDEPLOYING MICROSERVICE TLC2
20200702230328046 5 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2] UNDEPLOYING MICROSERVICE
20200702230328046 9 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] UNDEPLOYING CONTAINER CTLC2
20200702230328046 9 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC2] UNDEPLOYING CONTAINER FROM HOST HostTL2
20200702230328046 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2-moved] EXPOSING SERVICE TLC THROUGH INTERFACE tlcpl
20200702230328046 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2-moved] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702230328047 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2-moved] DEFINING REQUIRED SERVICE DevPL THROUGH INTERFACE devpl
20200702230328047 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2-moved] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702230328047 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2-moved] DEFINING REQUIRED SERVICE DevTL THROUGH INTERFACE devtl
20200702230328047 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC2-moved] DELEGATE ENDPOINT 8084 TO MICROSERVICE TLC2-moved ENDPOINT tlcpl
20200702230328047 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2-moved] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702230328047 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2-moved] PUBLIC BINDING devpl TO HOST HostPL2 ENDPOINT 18001
20200702230328047 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2-moved] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702230328047 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC2-moved] PUBLIC BINDING devtl TO HOST HostTL3 ENDPOINT 18021
20200702230328047 1 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702230328049 5 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (f0cd486674b24b5cadf9a8f486bcb72f) <<<
20200702230328049 6 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (c1643893d7e94918870f410537c42d11) <<<
20200702230328051 7 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL4] REMOVING ENDPOINT 18020
20200702230328051 7 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DEPLOYING CONTAINER CTLC4-moved
20200702230328051 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC4-moved] DEPLOYING CONTAINER CTLC4-moved IN HOST HostTL3
20200702230328051 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC4] REMOVING ENDPOINT 8084
20200702230328051 8 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL3] DELEGATE ENDPOINT 18020 TO CONTAINER CTLC4-moved ENDPOINT 8084
20200702230328051 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC4-moved] EXPOSING ENDPOINT 8084
20200702230328051 8 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC4-moved] DEPLOYING MICROSERVICE TLC4-moved
20200702230328051 9 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4-moved] DEPLOYING MICROSERVICE TLC4-moved IN CONTAINER CTLC4-moved
20200702230328051 9 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC4] UNDEPLOYING MICROSERVICE TLC4
20200702230328052 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4] UNDEPLOYING MICROSERVICE
20200702230328052 0 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL4] UNDEPLOYING CONTAINER CTLC4
20200702230328052 0 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC4] UNDEPLOYING CONTAINER FROM HOST HostTL4
20200702230328052 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4-moved] EXPOSING SERVICE TLC THROUGH INTERFACE tlcpl
20200702230328052 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4-moved] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702230328052 0 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4-moved] DEFINING REQUIRED SERVICE DevPL THROUGH INTERFACE devpl
20200702230328052 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4-moved] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702230328052 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4-moved] DEFINING REQUIRED SERVICE DevTL THROUGH INTERFACE devtl
20200702230328052 1 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CTLC4-moved] DELEGATE ENDPOINT 8084 TO MICROSERVICE TLC4-moved ENDPOINT tlcpl
20200702230328052 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4-moved] PUBLIC BINDING plwo TO HOST HostERP ENDPOINT 18030
20200702230328052 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4-moved] PUBLIC BINDING devpl TO HOST HostPL4 ENDPOINT 18001
20200702230328052 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4-moved] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702230328052 1 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [TLC4-moved] PUBLIC BINDING devtl TO HOST HostTL3 ENDPOINT 18021
20200702230328052 1 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702230328054 8 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (c1643893d7e94918870f410537c42d11) <<<
20200702230328054 9 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (09aae63dbf544d3393213037b846cf6d) <<<
20200702230328056 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1] UNBINDING plwo FROM HostERP::18030
20200702230328056 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2] UNBINDING plwo FROM HostERP::18030
20200702230328056 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3] UNBINDING plwo FROM HostERP::18030
20200702230328056 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL4] UNBINDING plwo FROM HostERP::18030
20200702230328056 8 [DEBUG] ...pek.lite.modules.impl.LoopModule: Adaptation aborted. Executing exception
20200702230328057 0 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (09aae63dbf544d3393213037b846cf6d) <<<
20200702230337091 8 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: PL4 => low
20200702230337091 9 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: PL4 => low
20200702230337091 9 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: PL3 => high
20200702230337092 0 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: PL3 => high
20200702230337092 1 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule LOAD-BALANCER> Asking for the next System Configuration
20200702230337092 7 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule LOAD-BALANCER> Request Planning (8267bd682396479a8eb16a63e63faa55)
20200702230337092 8 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (8267bd682396479a8eb16a63e63faa55) <<<
20200702230337092 8 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostPL1) => 1.3024691
20200702230337092 9 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostPL1) => 1.3024691
20200702230337093 0 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230337093 5 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (1f80dcfa632746d99a617577ab5e0f64)
20200702230337093 5 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostPL4) => 1.2708334
20200702230337093 6 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostPL4) => 1.2708334
```


```
20200702230370937 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230370941 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (678d7b1ccb7a46b59d23913150f13705)
20200702230370941 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostREW1) => 1.2307693
20200702230370941 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostREW1) => 1.2307693
20200702230370943 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230370946 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (1b491aed8cf64aaba70650f2c9acb3a8)
20200702230370947 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostTL1) => 1.1666666
20200702230370947 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostTL1) => 1.1666666
20200702230370948 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230370951 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (ff273b361e92423f8552fb2c5dc47eb9)
20200702230370952 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostTL2) => 0.0
20200702230370952 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostTL2) => 0.0
20200702230370953 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230370953 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] EXPOSING EDNPOINT 18021
20200702230370953 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] EXPOSING EDNPOINT 18020
20200702230370953 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL4] EXPOSING EDNPOINT 18021
20200702230370953 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL4] EXPOSING EDNPOINT 18020
20200702230370953 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL3] EXPOSING EDNPOINT 18000
20200702230370954 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] EXPOSING EDNPOINT 18001
20200702230370954 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (f7e00b840b4a49928caa7d17b0afeb1b)
20200702230370954 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL4] DEPLOYING CONTAINER PL3-1
20200702230370954 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-1] DEPLOYING CONTAINER PL3-1 IN HOST HostTL4
20200702230370954 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL2] DEPLOYING CONTAINER PL3-3
20200702230370955 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [PL3-3] DEPLOYING CONTAINER PL3-3 IN HOST HostPL2
20200702230370955 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostTL3) => 6.901961
20200702230370955 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostTL3) => 6.901961
20200702230370955 [DEBUG] ...pek.lite.modules.impl.LoopModule: Adaptation aborted. Executing exception
20200702230370956 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230370957 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (8267bd682396479a8eb16a63e63faa55) <<<
20200702230370958 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (1f80dcfa632746d99a617577ab5e0f64) <<<
20200702230370962 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (0081d4f59ec94ce38f7553211a61d377)
20200702230370963 [INFO ] ....lite.artifacts.components.Probe: Reporting measure: (Host HostERP) => 1.0833334
20200702230370963 [DEBUG] ...ite.artifacts.components.Monitor: Received measure: (Host HostERP) => 1.0833334
20200702230370964 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Asking for the next System Configuration
20200702230370969 [TRACE] ...ion.mapek.lite.impl.LoopResource: <Rule HOST-LOAD-LEVEL> Request Planning (85995b274ed94d39b471659981497058)
20200702230370977 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] DEPLOYING CONTAINER CDevTLPL2-moved-moved
20200702230370977 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL2-moved-moved] DEPLOYING CONTAINER CDevTLPL2-moved-moved IN HOST HostTL2
20200702230370978 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL2-moved] REMOVING ENDPOINT 8085
20200702230370978 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL2-moved-moved] EXPOSING ENDPOINT 8085
20200702230370978 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL2-moved-moved] DEPLOYING MICROSERVICE DevTLPL2-moved-moved
20200702230370978 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2-moved-moved] DEPLOYING MICROSERVICE DevTLPL2-moved-moved IN CONTAINER CDevTLPL2-
moved-moved
20200702230370979 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL2-moved] UNDEPLOYING MICROSERVICE DevTLPL2-moved
20200702230370979 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2-moved] UNDEPLOYING MICROSERVICE
20200702230370979 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL1] UNDEPLOYING CONTAINER CDevTLPL2-moved
20200702230370979 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL2-moved] UNDEPLOYING CONTAINER FROM HOST HostPL1
20200702230370980 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2-moved-moved] EXPOSING SERVICE DevTL THROUGH INTERFACE devtl
20200702230370980 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2-moved-moved] DEFINING REQUIRED SERVICE DigitalTwins THROUGH INTERFACE digTwins
20200702230370980 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2-moved-moved] DEFINING REQUIRED SERVICE ThingsREG THROUGH INTERFACE thingsReg
20200702230370980 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CDevTLPL2-moved-moved] DELEGATE ENDPOINT 8085 TO MICROSERVICE DevTLPL2-moved-moved
ENDPOINT devtl
20200702230370980 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2-moved-moved] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702230370981 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2-moved-moved] PUBLIC BINDING digTwins TO HOST HostERP ENDPOINT 18080
20200702230370981 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2-moved-moved] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702230370981 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [DevTLPL2-moved-moved] PUBLIC BINDING thingsReg TO HOST HostERP ENDPOINT 18081
20200702230370981 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702230380003 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (1f80dcfa632746d99a617577ab5e0f64) <<<
20200702230380004 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (678d7b1ccb7a46b59d23913150f13705) <<<
20200702230380032 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] EXPOSING EDNPOINT 18000
20200702230380033 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL4] REMOVING ENDPOINT 18000
20200702230380033 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] DEPLOYING CONTAINER CPL4-moved
20200702230380033 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL4-moved] DEPLOYING CONTAINER CPL4-moved IN HOST HostTL2
20200702230380034 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL4] REMOVING ENDPOINT 8080
20200702230380034 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] DELEGATE ENDPOINT 18000 TO CONTAINER CPL4-moved ENDPOINT 8080
20200702230380034 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL4-moved] EXPOSING ENDPOINT 8080
20200702230380034 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL4-moved] DEPLOYING MICROSERVICE PL4-moved
20200702230380034 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL4-moved] DEPLOYING MICROSERVICE PL4-moved IN CONTAINER CPL4-moved
20200702230380035 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL4] UNDEPLOYING MICROSERVICE PL4
20200702230380035 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL4] UNDEPLOYING MICROSERVICE
20200702230380036 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostPL4] UNDEPLOYING CONTAINER CPL4
20200702230380036 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL4] UNDEPLOYING CONTAINER FROM HOST HostPL4
20200702230380037 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL4-moved] EXPOSING SERVICE PL THROUGH INTERFACE pl
20200702230380037 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL4-moved] DEFINING REQUIRED SERVICE PLWO THROUGH INTERFACE plwo
20200702230380037 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL4-moved] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702230380037 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL4-moved] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702230380037 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CPL4-moved] DELEGATE ENDPOINT 8080 TO MICROSERVICE PL4-moved ENDPOINT pl
20200702230380037 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL4-moved] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702230380038 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL4-moved] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702230380038 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702230380066 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (678d7b1ccb7a46b59d23913150f13705) <<<
20200702230380067 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (1b491aed8cf64aaba70650f2c9acb3a8) <<<
20200702230380087 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] EXPOSING EDNPOINT 18010
20200702230380087 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW1] REMOVING ENDPOINT 18010
20200702230380087 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] DEPLOYING CONTAINER CREW1-moved
20200702230380088 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CREW1-moved] DEPLOYING CONTAINER CREW1-moved IN HOST HostTL2
20200702230380088 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CREW1] REMOVING ENDPOINT 8082
20200702230380088 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostTL2] DELEGATE ENDPOINT 18010 TO CONTAINER CREW1-moved ENDPOINT 8082
20200702230380088 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CREW1-moved] EXPOSING ENDPOINT 8082
20200702230380088 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CREW1-moved] DEPLOYING MICROSERVICE REW1-moved
20200702230380089 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW1-moved] DEPLOYING MICROSERVICE REW1-moved IN CONTAINER CREW1-moved
20200702230380089 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CREW1] UNDEPLOYING MICROSERVICE REW1
20200702230380089 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW1] UNDEPLOYING MICROSERVICE
20200702230380090 [INFO ] ....lite.ARM.impl.AdaptiveReadyHost: [HostREW1] UNDEPLOYING CONTAINER CREW1
20200702230380092 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CREW1] UNDEPLOYING CONTAINER FROM HOST HostREW1
20200702230380093 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW1-moved] EXPOSING SERVICE REW THROUGH INTERFACE rew
20200702230380093 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW1-moved] DEFINING REQUIRED SERVICE Transf THROUGH INTERFACE transf
20200702230380093 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW1-moved] DEFINING REQUIRED SERVICE REWWO THROUGH INTERFACE rewwo
20200702230380093 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW1-moved] DEFINING REQUIRED SERVICE Stock THROUGH INTERFACE stock
20200702230380093 [INFO ] ....ARM.impl.AdaptiveReadyContainer: [CREW1-moved] DELEGATE ENDPOINT 8082 TO MICROSERVICE REW1-moved ENDPOINT rew
20200702230380093 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW1-moved] PUBLIC BINDING transf TO HOST HostERP ENDPOINT 18060
20200702230380093 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW1-moved] PUBLIC BINDING rewwo TO HOST HostERP ENDPOINT 18031
20200702230380093 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [REW1-moved] PUBLIC BINDING stock TO HOST HostERP ENDPOINT 18070
20200702230380093 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702230380119 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (1b491aed8cf64aaba70650f2c9acb3a8) <<<
20200702230380119 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (ff273b361e92423f8552fb2c5dc47eb9) <<<
20200702230380137 [DEBUG] ...pek.lite.modules.impl.LoopModule: Adaptation aborted. Executing exception
20200702230380138 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (ff273b361e92423f8552fb2c5dc47eb9) <<<
20200702230380138 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (f7e00b840b4a49928caa7d17b0afeb1b) <<<
```

```
20200702230338016 2 [INFO ] ...ite.modules.impl.ExecutingModule: Adaptation success. Updating the current configuration ...
20200702230338019 3 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (f7e00b840b4a49928caa7d17b0afeb1b) <<<
20200702230338019 3 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (0081d4f59ec94ce38f7553211a61d377) <<<
20200702230338021 5 [DEBUG] ...pek.lite.modules.impl.LoopModule: Adaptation aborted. Executing exception
20200702230338021 6 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (0081d4f59ec94ce38f7553211a61d377) <<<
20200702230338021 6 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## STARTING NEW ADAPTATION PROCESS (85995b274ed94d39b471659981497058) <<<
20200702230338023 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL1] UNBINDING plwo FROM HostERP::18030
20200702230338023 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL2] UNBINDING plwo FROM HostERP::18030
20200702230338023 8 [INFO ] ...M.impl.AdaptiveReadyMicroservice: [PL3] UNBINDING plwo FROM HostERP::18030
20200702230338023 8 [DEBUG] ...pek.lite.modules.impl.LoopModule: Adaptation aborted. Executing exception
20200702230338023 9 [INFO ] ...ite.modules.impl.AnalyzingModule:    >>> ## ADAPTATION PROCESS ENDED (85995b274ed94d39b471659981497058) <<<

simulator pause
```