

SAP Hybris retires YaaS.io – where do SAP microservices go from here?



By **Dick Hirsch** May 21, 2018

SUMMARY:

SAP has announced the end for the pioneering YaaS.io. But as Dick Hirsch advises – don't send flowers. SAP has more plans for microservices in the SAP Cloud Platform mix.



2 Comments

After **an unexpected illness in January of this year**, Hybris' YaaS.io passed away ("retired") on May 1. YaaS.io – otherwise known as **Hybris' Commerce as a Service** – is survived by an older sibling – **SAP Cloud Platform (SCP)**.

YaaS.io was **born with great fanfare in 2014** and gained in importance in 2015 with a focus on **Business Services in**

SAP's Cloud offerings. Its presence will be missed by many who were impressed by its early support of microservices and its marketplace.



In lieu of flowers

Although I'm blogging less now, I'm still interested in the evolution of the **SAP Cloud Platform** (SAP's PaaS) – in particular, the **services that are available to develop applications on the platform**. In my opinion, those services are an important differentiating factor for the offering – for example, there are many blockchain or IoT offerings in the Cloud but few companies can offer the combination of more technically-

focused services (SAP Leonardo, etc.) with the process-related services that originate in the decades of experience that SAP has in this area.

As I recently checked the list of SCP business services, I saw the description of **YaaS** and fondly recalled my **last blog series of the Hybris offering** from 2017 in which I analyzed in detail how YaaS.io was an excellent example of agile- and microservice-based development. As I started looking at the **YaaS developer site**, I noticed **a strange lack of activity/releases** in comparison to last year. I was curious to find out the cause and found a **blog from January 9** with the sad details.

Note: *the original blog has been removed but much of the content is duplicated in the official retirement notice. This newer blog also has more details.*

“ *As announced in the YaaS.io DevPortal Blog on January 9, 2018, SAP Cloud Platform enabled access to all eligible SAP Cloud Platform services with a consumption-based model. You now have the flexibility to use and pay for consumed services within a prepaid cloud credit balance. You can find more details at <https://cloudplatform.sap.com/pricing.html>.*

Due to this new offering and the YaaS 1.0 sunseting, the YaaS.io marketplaces will be retired at the end of April, 2018. Parts of the YaaS.io offering will migrate to SAP Cloud Platform, and other services will not be available anymore. Commercial subscriptions to any YaaS.io packages will be cancelled on January 25th and, after the 90-days notice, they will end on April 30, 2018. The beta marketplace and its subscriptions will be closed at the end of April, 2018.

Further impact:

- The YaaS Dev Portal Blog shuts down at the end of April 2018.*
- Invoices generated for your organization are available for download until the end of April 2018. After that date, you will not be able to access the invoices through the YaaS Builder.*
- The Hybris Experts branch for YaaS will be available in the read-only mode.*

Other content from YaaS.io remains available in the read-only mode until further notice. [devportal.yaas.io]

I must admit that this announcement was a bit of shock and I was unsure exactly what it meant. Rereading the **original blog's title** “Replacement of YaaS.io Marketplaces by SAP Cloud Platform” made things a bit clearer. In response to an enquiry from a YaaS developer, a Hybris forum post provided even more details regarding the impact of the sunseting of specific YaaS services.

1. You can pick the right service for persistence offered at SAP Cloud Platform when you browse to <https://cloudplatform.sap.com/dmp/capabilities/us> > “Data & Storage”. To store media, you can use e.g. the “SAP Cloud Platform Document Service” <https://cloudplatform.sap.com/dmp/capabilities/us/product/SAP-Cloud-Platform-Document-Service/5a039b46-8bdd-469e-9613-1d09d8c5cebe>
2. Neither the packages currently summarized underneath “Commerce as a Service” (Customer Accounts, Product Content, Order Management, Site Management, Cart, Checkout, Coupon Management) nor the E-mail package will have successors or substitutes at the SAP Cloud Platform offering at this point in time.
[experts.hybris.com]

Before the YaaS Dev Portal Blog was shutdown, specific YaaS.io services (for example, [SAP Hybris Profile Packages](#) and [SAP Hybris Service Engagement Center](#)) as well as documentation-related projects (for example, [YaaS Bites](#)) provided additional information regarding their retirement. Some migration support was provided (for example, [SAP Hybris Profile](#)) but some developers using other YaaS.io services had difficulty figuring out what options were available to them (for example, [here is one developer trying to figure out how to use the Loyalty Program](#) who was provided a solution to migrate to the [SAP Hybris Loyalty solution](#)).

What is a bit strange is that there are still releases coming out for some packages (for example, [for the Hybris Profile service](#)) despite the fact that the YaaS.io marketplace no longer exists and the Hybris Profile Service was retired ([see the blog from January 9 entitled “Retirement of SAP Hybris Profile Packages”](#)).

The reason that this activity is still ongoing is that the underlying YaaS.io platform still exists in order to support those remaining customers who have yet to migrate to SCP. There is also limited development work going on to support those few Hybris products still using the platform / infrastructure.

Business services and microservices

One of the reasons that YaaS.io and its service marketplace was initially an important component of SAP’s Cloud strategy was its focus on business services “[to encapsulate best-practice business functions and expose these functions as APIs](#)”. When YaaS was released in 2014, there were not very many business services that were available for SAP’s Cloud offerings. In the meantime, there are many [such APIs](#) that are available that cover a variety of SAP’s SaaS applications ([S4/Hana Cloud](#), [FieldGlass](#), etc.) as well as OnPremise deployments ([Banking](#), etc.).

A more detailed look at these services is interesting. For example, Concur has a variety of REST-based APIs including [Concur Expense](#), [Concur Invoice](#), [Concur Request](#). As you start to look at the particular offerings, however, you will usually see that [all APIs are from the same version](#) and have the same last modification date. My assumption is that the underlying APIs probably originate from a largely monolithic application – albeit one available in the cloud – with a single release/deployment schedule.

If you are using one of SAP's SaaS applications, you have a wide range of APIs that are available to extend your application (I'm disregarding [indirect access licensing](#) aspects to keep things a bit more simple). Indeed, the presence of such APIs in combination with SCP as an extension platform is one of the more important scenarios for such SaaS offers.

I highly doubt, however, that you can select a single API for one of these SaaS applications and pay for it separately using SAP's new consumption-based pricing. This distinction is also evident in SCP's [Service Catalog](#) with the difference being between "Services" and "API Packages."

The bundled APIs are associated with a single SaaS application to which the customer already has access ("Get access to and test all of the API packages that come with your SAP SuccessFactors solution," "Use our API packages to extend your SAP S/4HANA solution," etc.). Usually, such bundled offerings are available via SAP's ever-growing [API Hub](#).

One of the reasons that YaaS was unique was not only the fact that it provided business services as APIs but rather the underlying architecture of such services. YaaS was a microservice-based architecture that provided these APIs:

“YaaS is SAP Hybris' entry into the emerging business application IT model of microservices which provides an additional, innovative cloud based solution deployment option. This goes beyond the traditional on-premise (up-front purchase) and public cloud application (multi-tenant architecture, multi-year leasing, annual contract values) models. Based on a concept of developing applications as a suite of small services, each microservice runs its own processes and communicating with lightweight mechanisms (often HTTP REST API, stateless). Microservices are built around business capabilities and are independently deployable. [[yaas.io](#)]

I discussed the importance of this architecture [last year in my blog about YaaS and microservices](#). The difference between YaaS and other Hybris offers is described in [a Hybris forum post](#): "SAP Hybris Commerce is claudified (sic) and plans to gradually enable a microservices architecture. Commerce as a service, as known from YaaS.io, will be discontinued." One important advantage of this microservice-based architecture was that each [CaaS microservice had a specific consumption model and could be purchased separately](#).

Earlier this year, SAP announced that it was shifting to **a consumption-based pricing** for some cloud solutions. This shift is evident in many of the technical SCP services (for example, **Forms by Adobe**, **Internet of Things**, etc.) – each of which has a different pricing model. Most of these services also have **their own release cycle** which demonstrates a certain release/development agility.

What is curious is the lack of “business services” in the consumption-based SCP Service catalog – there are **only two** – “SAP Data Quality Management” and “SAP Localization Hub, tax service.” Regarding the “**Tax Service**”, there is a **similar API in YaaS** and the service in the SCP service catalog **hasn’t been updated in a while**. Despite the fact the CaaS microservices would be probably bring useful business-related functionality to the SCP catalog, it appears that they will **not be migrated to SCP**.

I plan to be on the ground at Sapphire Now 2018 in Orlando, where I will seek clarifications on the reasons for the demise of YaaS.io – and where SAP intends to go from here. Watch this space.

Note: *Hybris Commerce content is still available **via the API Hub** but the API description point to YaaS.io so I’m not exactly sure if they are still valid or not.*

My take

Consolidation is important for SAP’s Cloud-related activities in that there are still many moving parts/various solutions (although things have definitely improved over the last few years) and a simplified architecture improves brand messaging (for example, the “Run Simple” campaign from 2015) as well as helping reduce SAP’s associated development and operational costs. A **recent Constellation Research blog** from fellow blogger Holger Mueller shows that the long-discussed conversion process for SAP’s different SaaS application might finally becoming reality.

“ *Since SAP put the ‘5 sisters’ (Ariba, Concur, Fieldglass, Hybris, SuccessFactors) under the leadership of Rob Enslin, it’s clear that the long-stalled technology conversion of these five business towards the SAP technology stack was about to happen. More specifically it means the long talked, not much executed adoption of SAP HANA as the database, the adoption of SAP Cloud Platform as a PaaS, the use of SAP Analytics as reporting and Analytics platform, the usage of UX elements coming from Fiori (here the SAP Cockpit) and of course SAP Leonardo for machine learning and more.*
[constellationr.com]

In the case of the ‘5 sisters’, the underlying technology appears to be consolidated, but the associated business content/services will remain largely unchanged. Although it was

promoted originally as a broader business service platform, YaaS.io was largely a niche product focusing on commerce-related topics (yes, I know there were exceptions, such as [Earth Observation Analysis](#)). Consolidation in the case of YaaS, however, appears to be more associated with the adoption of more fundamental architectural characteristics (consumption-based API, marketplace, CloudFoundry, etc) rather than a migration of the associated microservices.

SCP, however, didn't just copy those YaaS.io characteristics – important enhancements were made as well. For example, the consumption model used by YaaS.io was addressed more towards the individual developer using a credit card to purchase individual YaaS.io APIs/services. SCP's consumption model is more nuanced and better reflects the requirements (contractual, procurement, predictability, etc) that most SAP customers have regarding the purchase of cloud and other enterprise software services.

It is important to pay homage to the innovative character of YaaS.io. In my opinion, YaaS was often an early adopter of PaaS-related technology (for example, CloudFoundry) and the agile development methodology (for example, microservices). This characteristic might be associated with the relatively small size of the YaaS dev team (at least, in comparison to SCP), its smaller customer base as well as the niche character of its offering.

For example, the YaaS.io marketplace had always used a consumption-based model and, to some degree, was a test-bed for the [transition of SCP to the consumption-based model](#). The [SCP Service Catalog](#) is similar to the YaaS Marketplace ([which is no longer available](#)). Another [example is the use of CloudFoundry in YaaS that emerged in 2014](#) – before the same architecture was present in SCP (although my assumption is that the SCP team was aware of CloudFoundry (CF) and was probably exploring its potential in various internal PoCs – in particular, how CF-based usage scenarios would fit into a strategy that was at that juncture largely based on Neo).

Note: *The technical characteristics that were adopted early by YaaS weren't unique to the Hybris' platform but reflected much broader trends in IT. Thus, even if YaaS hadn't existed, SCP probably would have developed in this direction. Yet, my assumption is that the presence of practical experience regarding such activities proved valuable to the broader SAP Cloud organization in its evolution.*

Yaas.io's [retirement blog](#) closes with the following statement: "The spirit of YaaS to transform enterprises, in the microservice way, persists. Stay tuned for further announcements on the transformation of YaaS itself." As a [recent Hybris job offer](#) reveals, it appears the Hybris team is already working on something new.

Our part of the organization is responsible for developing and providing the SAP Hybris Service Factory, a central extension and integration framework which covers the

“ *complete SAP Hybris product portfolio. The SAP Hybris Service Factory provides the capabilities to build fully distributed, highly scalable, microservices based custom solutions through a cloud-native approach, enabling software development or IT organizations to be more efficient, effective with an increasing agility.*

This description is pretty vague but a different **event description** on the topic reveals that it appears to be associated with *Function as a Service* and serverless technology – in particular **Kubeless**– and that the **SCP and Hybris teams are working together on the topic**.

“ *Serverless Technologies enabling developers to build applications by configuration. The serverside implementation relies on existing services (e.g. Databases, API-Proxies, etc.) and lambda functions deployed on a Function as a Service (FaaS) Platform. All major cloud providers are offering vendor specific FaaS solutions. To reduce dependencies (sic) to cloud providers and enable deployments in the private cloud, alternatives based on kubernetes are currently developed. SAP Cloud Platform and SAP Hybris are contributing to Kubeless, a kubernetes native Serverless Framework. In the talk, we will have a look in kubeless and how it is used in SAP Hybris Service Factory. [[meetup.com](https://www.meetup.com)]*

It appears that this new framework may be called “**SAP Cloud Platform Extension Factory**” with a project name “**Kyma**”.

YaaS.io has passed away but it appears that something just as exciting is currently underway and the Hybris and SCP teams are actively collaborating in this area. There is even a recent **DJ Adams blog** on the topic to prove just how cool it is....

End note: prior to the publication of this post, I spoke with SAP to confirm the relevant details. We'll share future updates on SAP's cloud strategy after Sapphire Now in June.

Image credit - Sherlock © LuckyImages - Fotolia.com

Disclosure - SAP is a diginomica premier partner.

Read more on: Cloud ERP financials and supply chain |
Cloud platforms - infrastructure and architecture |
DevOps NoSQL and the open source stack