

Question

[Utsavi Gajjar](#) · Apr 15, 2021

Best practices for implementing REST Interfaces in an IRIS Production

Hello Everyone,

We are currently in process of implementing REST APIs using IRIS and we are also looking at using InterSystems API Manager.

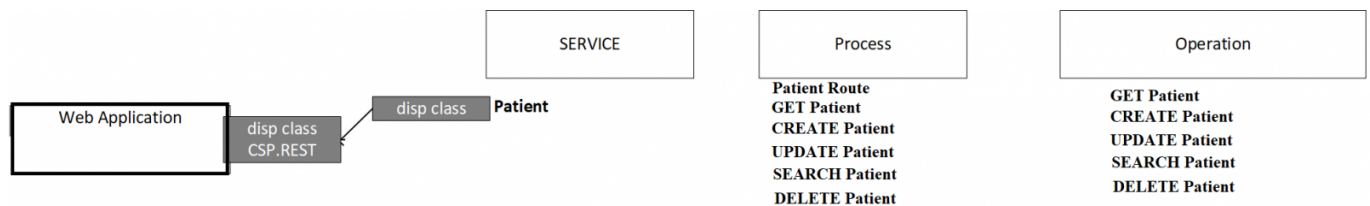
Our aim is to implement a Microservices Architecture where Services are small in size, bounded by contexts, autonomously developed, and independently deployable.

We are following a spec first approach where we are first defining the API specs into Swagger Hub and using IRIS API Management Service to build the REST classes i.e. the Specification class, Dispatch class and Implementation Class.

Methods in Implementation class invoke a Business service in IRIS Production and then a Business process and an Operation. We are creating a Business Service per resource e.g. "Patient" but we want to keep the Business processes and Operations as discrete as possible hence we are creating a BP and BO per Action on a Resource.

So for example for a "Patient" resource if the possible actions are Add, Update, Delete, Read and Search. In the IRIS production, we have one Business Service, 6 Business Processes (1 for each Action plus one for routing) and 5 Business Operations. (Shown in the screenshot below).

This would mean that if we have 50 different resources and each resource had these basic 5 Actions, we'd end up with 250 Business processes and 250 Business Operations. Our assumption is that the concern around spawning too many underlying processes (for each BP and BO) will be addressed by relying on Actor Pool for low traffic interfaces and allocating more resources/processes to high traffic ones.



Can I please ask for advise on whether this is the right approach or not? If not, what are the concerns and how can we address those and what is the better approach?

Thanks heaps.

Utsavi

[#Microservices](#) [#REST API](#) [#InterSystems IRIS for Health](#)

Product version: IRIS 2020.1

Source URL: <https://community.intersystems.com/post/best-practices-implementing-rest-interfaces-iris-production>

