REST API WITH MEISTER 2.0

MEISTER V2.0 ORCHESTRATION ENGINE

Andre Rosenthal
Gateway Architects, LLC
2600 Dallas Parkway suite # 600, Frisco, TX

Table of Contents

Product Document

Document Name:	Rest API with Meister 2.0
Document ID:	MV2INSP0002
Document Owner:	Andre Rosenthal
Document Version:	1.0
Document Date:	Saturday, 24 August 2019
Document Status:	Draft

Introduction

As Meister 2.0 provides a unique integration via pure Rest services, the next natural feature is to provide a common API that is naturally fitting for all situations. Consider the swagger model: we would need to promote a process which is part of Meister and modelled on the syntax and processes expected by Swagger in order to manipulate the needs of SAP while maintaining the flexibility of a Web API.

Swagger Integration

We will borrow our mapping function defined at the Integration with Meister 2.0 document:

$$g\left(\sum f_i\left(x\right)\right) \mid \exists\, x\ in\ X\ and\ y\ in\ Y \rightarrow\ \exists\, map\ M(f,g)\ defining\ a\ transition\ from\ t\ to\ g.$$

We called this function $g()$ before as the Integrator function, or in simple form, the orchestration engine behind the mapping of discrete data from different systems. We could therefore create another function as above such that the transition portion would be from a Swagger modelled set of APIs to the uniqueness of SAP.

Leveraging the Swagger model

Meister 2.0 operates in conjunction with Swagger and provides the direct pipe to Swagger via Yaml and exposes Meister endpoints as simple Web API calls. From the swagger interface, calls executed via the Web API would end up as a simple Meister 2.0 endpoint call directly through Meister.

The Web API

In terms of Swagger the initiator of the call is just a simple Rest client, as such the initial set of APIs are the normalization of the Meister endpoint call. The Meister SDK provides the enveloping whereupon a call from SDK is transformed into the Yaml equivalent of the Web API. That means in fact that all necessary support to create the Web API code from Swagger is published directly by the SDK into Swagger as a process driven by the SDK.