



COMPILER, COMPRESSION AND PERFORMANCE

MEISTER SUITE OF WHITE PAPERS

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





Table of Contents

<i>Json to ABAP runtime compiler</i>	3
<hr/> <hr/>	
<i>Compression and performance</i>	3
<hr/> <hr/>	
<i>Protocol Compression</i>	3
<hr/> <hr/>	
<i>Byte Compression</i>	3



Product Document

Document Name:	Compression
Document ID:	Mv2whtper0001
Document Owner:	Andre Rosenthal
Document Version:	1.0
Document Date:	Monday, 26 August 2019
Document Status:	Draft



Json to ABAP runtime compiler

Meister is an in-memory cross-compiler for Json and ABAP. It receives the request in json from the sender and compiles the json into ABAP to be used by the SAP ABAP stack. As the execution ends, Meister gets the SAP ABAP stack response, the compiles it into json to be returned to the sender.

With this unique feature, there is no need for the caller of the transaction to be exposed to OData syntax and the unique entity set methodology anymore. Given that with OData and entity set discreteness, it is necessary for the UX layer to be exposed to business logic needed to link the discrete entities into a logical unit. Since Meister hides all OData complexity within the compiler itself, the UX receives a comprehensive response using SSC which already has all defined dependencies between the entities and/or entity sets in a single call.

Compression and performance

To compensate for the larger payloads used by SSC, Meister uses two techniques for compression and performance:

Protocol Compression

Protocol compression is enforced by Meister which achieves compression factors of 40% on average. Using Outside-in processes the payload is returned via GET and is bound by the capacity of the browser or web Rest library being used by the UX.

Byte Compression

Meister adds to the Protocol Compression a byte compression logic that further reduces the payload returned to an average of 80%. Using Inside-out processes the payload is returned via POST and therefore is only bound by the memory at the UX layer.