



Dominique Alfermann, Stefan Hartmann, Benedikt Engel

SAP® HANA Advanced Modeling

- ▶ Data modeling guidelines and common test approaches
- ▶ Information view performance optimization
- ▶ Modular solutions to complex requirements
- ▶ Best practices and recommendations

Table of Contents

| | |
|--|------------|
| Preface | 5 |
| 1 Introduction | 9 |
| 1.1 Intention | 9 |
| 1.2 Objective | 10 |
| 1.3 In and out of scope | 10 |
| 1.4 Content | 11 |
| 1.5 Data model definition and terms | 13 |
| 2 Case study and hardware setup | 15 |
| 2.1 The demand for SAP HANA native solutions | 15 |
| 2.2 SAP HANA native project introduction | 17 |
| 2.3 System landscape and configuration | 25 |
| 2.4 Test approach | 26 |
| 2.5 Summary and conclusion | 30 |
| 3 Create the data model | 31 |
| 3.1 Persistency versus virtualization | 31 |
| 3.2 Modeling approaches | 41 |
| 3.3 Engine utilization | 51 |
| 3.4 PlanViz | 55 |
| 3.5 Test scenarios | 63 |
| 4 High-performing information views | 85 |
| 4.1 Attribute view | 86 |
| 4.2 Analytic view | 96 |
| 4.3 Calculation views | 130 |
| 5 Advanced modeling techniques | 189 |
| 5.1 Development perspective | 189 |
| 5.2 Core Data Services | 190 |

| | | |
|----------|---|------------|
| 5.3 | Challenges in SAP HANA modeling | 195 |
| 5.4 | Development methods | 221 |
| 6 | Best practices and recommendations | 231 |
| 6.1 | SAP view strategy | 231 |
| 6.2 | SAP HANA table design | 232 |
| 6.3 | Joins | 233 |
| 6.4 | Calculations | 238 |
| 6.5 | Currency and unit conversion | 241 |
| 6.6 | Filters | 244 |
| 6.7 | General SAP HANA design aspects | 247 |
| 6.8 | Summary | 251 |
| 7 | Operational reporting on SAP HANA | 253 |
| 7.1 | Operational versus strategic reporting | 253 |
| 7.2 | Specifics of operational reporting | 255 |
| 7.3 | Operational reporting scenario | 264 |
| 7.4 | Operational reporting optimization | 267 |
| 7.5 | SAP HANA Live | 270 |
| 7.6 | Summary | 278 |
| 8 | Conclusions and outlook | 281 |
| 8.1 | Conclusions | 281 |
| 8.2 | Outlook | 283 |
| A | About the Authors | 290 |
| B | Index | 293 |
| C | Disclaimer | 296 |

2 Case study and hardware setup

In this chapter, we use a case study to explain how to implement an SAP HANA native solution. We describe the as-is situation, including known issues with the current business warehouse, develop an entity relationship model, and propose a system landscape. This analysis will serve as the basis for the planned solution. The chapter concludes with a high-level view of our test approach.

2.1 The demand for SAP HANA native solutions

Through our daily project work with clients in recent years, we have noticed an increasing demand for real-time reporting and flexible data warehouse solutions. The need for real-time reporting in areas such as logistics, planning processes, market basket analyses, and next best action recommendations emphasizes the numerous use cases. These examples underscore the demand for a wide range of reporting solutions. Given the reporting requirements, clients are likely to invest in these types of technologies. Furthermore, many research facilities regularly analyze reporting needs and user experience. Gartner, Inc. recently published an analysis stating that by using real-time reporting in consumer businesses, for example by sales representatives leveraging data on a daily basis, revenue increased by up to 17%. The corresponding analysis can be found in the article *Fast-Fashion Retailer Boosts Sales With Real-Time Big Data Analysis* by Alicia Fiorletta in Retail TouchPoints.¹

Based on the fashion retailer Bohme, the article analyzes and highlights the benefits of real-time reporting.² According to the article, the retailer achieved a 15% increase in sales shortly after implementing a real-time reporting solution. The company's employees had to deliver an unreasonable amount of work in order to handle warehouse stock and maintain operations. The article states that "tracking and analyzing the sheer

1 <http://www.retailtouchpoints.com/topics/store-operations/fast-fashion-retailer-boosts-sales-with-real-time-big-data-analysis> ⇒1

² <http://www.risual.com/retail>

variety and velocity of data became too cumbersome.”³ Once the company implemented a real-time reporting solution, they achieved a turnaround. Using different visualization types, such as dashboards to report relevant KPIs etc., improved collaboration between the shop floor (production team) and management significantly.

The development of real-time data for sales representatives had a positive impact on the company’s revenue.

Best practice emphasizes the importance of real-time reporting for the effective management of companies, as well as for their daily business. Above all, it shows that there is a vast market for real-time data analysis that can be used for different industries and approaches.

Since SAP announced its new SAP HANA technology, these topics have received higher visibility within companies already using SAP, as well as those considering implementing SAP in their IT environment. However, whether or not a company should commit to SAP HANA, particularly an SAP HANA native implementation, is a question that is occupying many decision makers.

SAP Societas Europaea (SE) dedicated itself to this question in a concerted effort with its customers and analyzed their motivations for choosing an SAP HANA native solution. They identified the following key aspects (see Figure 2.1).

Speed is one of the main arguments for an SAP HANA native implementation. This was outlined by Mr. Akihiko Nakamura, the Corporate Senior Vice President of the Nomura Research Institute: *“Now and in the future, speed is the key to adapting to an ever-changing business environment. The speed SAP HANA enables is sudden and significant, and has the potential to transform entire business models.”*⁴

In addition, agility, as described by SAP, enables real-time interactions across a company’s value chain.

³ CEO BOHME <http://www.retailtouchpoints.com/topics/store-operations/fast-fashion-retailer-boosts-sales-with-real-time-big-data-analysis> ⇒2

⁴ SAP Top 10 Reason <http://www.tyconz.com/more/Top10ReasonsCustomersChooseSAPHANA.pdf> ⇒3

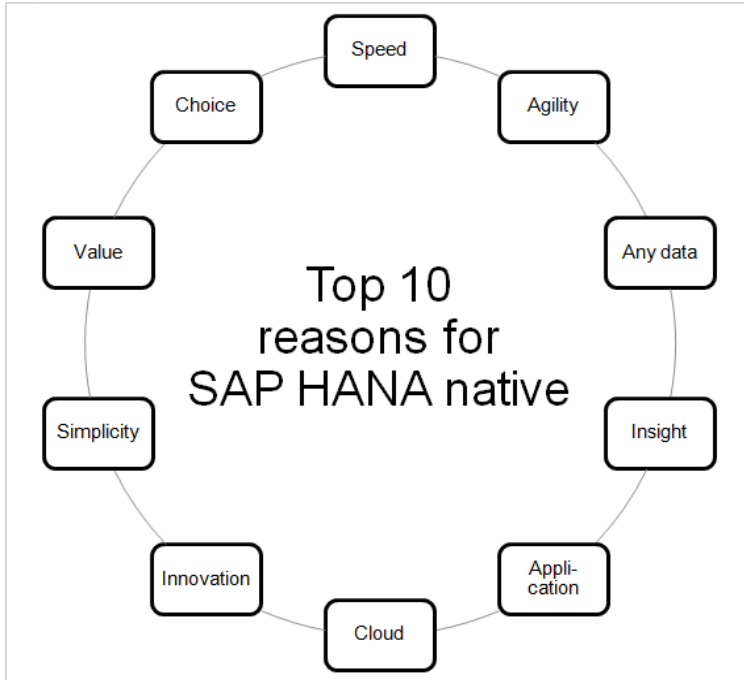


Figure 2.1: Top 10 reasons for SAP HANA native

Another argument for SAP HANA native is the ability to build interfaces for structured and unstructured data. In other words, it enables various heterogeneous data sources to be connected. One of the main benefits of this solution is *insight*, enabling faster response times and consequently, better planning and predictions. Finally, *simplicity* implies the ability to quickly and easily create data flows as long as complex logic is not required. Other aspects such as costs and utilization have to be considered, as well as the return on investment (ROI). SAP clients keep asking questions such as *why should I use SAP HANA native and not some other in-memory solution?*

2.2 SAP HANA native project introduction

The customer in our case study is a well-known retail company. Their customer analytics department is tasked with focusing on customer behavior. Their analysis looks at a variety of questions, such as the average purchase price per city, repeat customer profiles, and products that sell well.

Our retail company is the market leader in analyzing data and predicting future shopping behavior and trends. They want to sustain a solid market share growth and therefore, the business department intends to implement a real-time reporting solution. In the CIO's opinion, SAP HANA native provides a suitable approach for their end users in the business department.

The IT department is often in conflict with the business department regarding the required reporting processes. The business department intends to analyze the buying patterns of consumers, as brands often invest in marketing campaigns such as discounts and bonus programs in order to incentivize customers to buy their products. However, the most valuable clients are not the one-time purchasers but those buying the brand more frequently. This is actually the most challenging target. For that reason, the business department wants to improve the quality of their analysis in order to increase their performance.

The IT Director decides to set up a new project to implement SAP HANA native in order to provide the business department with the foundation for real-time reporting.

Therefore, the IT Director introduces a simple process grouped into the three clusters noted in Figure 2.2.

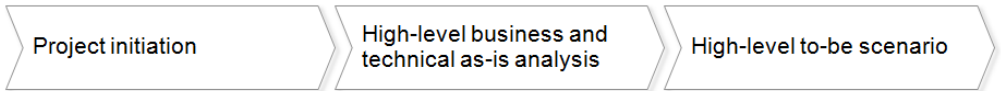


Figure 2.2: Project process

The project initiation represents the project's vision and provides an overall picture of the project. Of course, the IT Director has to convince each stakeholder that he or she is a part of the project and supports the overall vision.

Based on an employee survey and data analysis, the as-is situation is evaluated and the requirements are defined. The final future scenario is described in an entity relationship model (ERM) which is considered the foundation for the SAP HANA native solution.

2.2.1 Project initiation

In our case study, the IT Director wants to provide the business department with a best-in-class reporting solution for their daily work. The aim of this project is to replace the existing SAP business warehouse (BW) with the SAP HANA native solution.

Figure 2.3 shows the current system landscape at a high level. We will cover the solution in more detail when we discuss the future scenario.

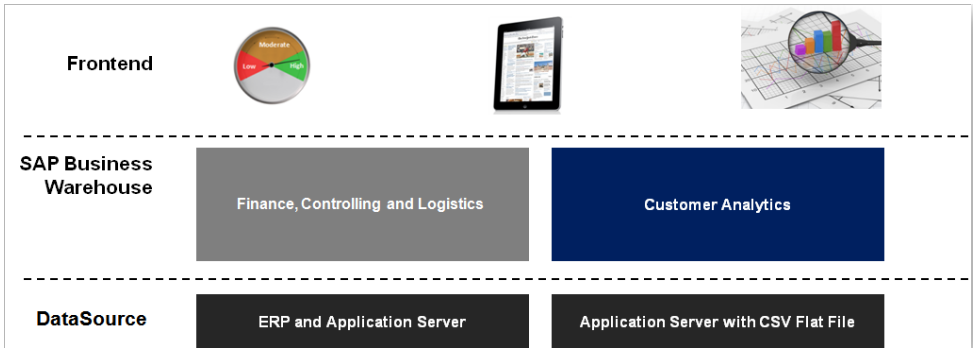


Figure 2.3: High-level SAP BW architecture

The relevant aspect of this project will be the customer analytics data model and the underlying source data. This data model will be reconstructed and adapted to a solution based on SAP HANA native.

The customer analytics will be provided in the SAP HANA Cloud and the new system landscape will be built from scratch. The selected approach should be stable for the next couple of years. To obtain the buy-in of all employees involved, the IT Director demonstrates the strengths of the solution using an SAP HANA native prototype developed within Cloud Services by an IT consulting company. As all stakeholders agree to this solution and the project scope has been defined accordingly, the project can now start.

In the next section, we will take a closer look at the as-is situation. This includes the reporting environment as well as the data itself which serves as the basis for reporting.

B Index

A

Aggregation 42, 186, 187, 188, 192, 193, 209, 241
Aggregation node 127, 149, 174, 188, 192, 199, 227
Agility 17, 212, 238
Analytic view **42**, 47, **91**, 219, 223
Attribute view **42**, 51, **82**, 219, 233
Average aggregation 186, 188

C

Calculated column 86, 87, 107, 109, 110, 111, 120, 121, 127, 130, 146, 148, 193, 227, 228
Calculation before aggregation 107, 188, 226, 227
calculation engine **51**
Calculation engine 52, 53, 59
Calculation view **42**, 52, **125**, 219, 237
 Scripted 52, 163, 165, **166**, 168, 208
Cardinality 84, 85, 103, 104
CE function 163, 164, 166
Constant column 138, 140, 234
Constraint filter 99, 137, 143
Core Data Services 180, 237
Counter **118**, 119, 153, 154, 193
Critical path 58
Currency conversion 111, 113, 114, 148, 149, 229, 232

D

Data duplication 170, 172, 204
Data recovery 36
Data replication **44**
Data type 65, **66**, **67**, 80, 86, 107, 114, **184**, 221
Date function 147
Denormalization 62
Development perspective **179**, 180, 210

E

EIM **34**, 35, 36
Engine utilization 49, 56, 59
ETL 37, 48
 Transformation 34, 37, 39, 46, 245
Exception aggregation 155, 156, **194**
Execution plan **53**, 56

F

Filter 90, 91, 97, 99, 102, 125, 127, 135, 137, 138, 158, 173, 232, 234
 Pushdown 172, 173
Foreign key **74**, **76**, 77, 183, 220

G

Governance 43, 216

H

Hidden column 123, 124, 161, 162
Hierarchy 150, 151, **152**, **153**
Historization 244
 Historical data 32, 33, 217, 245
 Historical truth 33, 204
History table 33, **206**

I

Index server 50
Information view **14**
Input parameter 54, 90, 91, 99, 102, 137, 138, 158, 160, 165, 166
 Mapping **55**

J

Join 42, 51, 68, 83, 85, 92, 103, 105, 140, 143, 145, 221, 226
 Referential 85, 95, 107, 225
 Star join 141, 143, 219, 221
 Temporal 121, 122, 123

K

Keep flag 155, **156**, **192**
Key attribute 88, 89

L

Load performance 77, **237**
LSA++ **32**, 244

M

Master data 33, 249
 Time-dependent 33, **204**
Memory consumption 41, 63, 72, 73, 74, 220, 258
Metrics 111, 113, 146, 148
Moving average 195
ulti-temperature concept **217**

O

OLAP engine **51**, 59, 224
Operational reporting 41, **241**, 243, 252, 256, 259

P

Parallel development **210**
Partitioning 68, 70
Persistency **31**, 32, 37, 43, 48
PlanViz **53**, 56, 59
Primary key 68, **72**, 74, 77, 79, 181, 220
Private view **215**, 238, **260**
Procedure 43, 47, 160, **168**, 170, 211
Projection node 125, 130, 133, 134, 135, 149, 170, 172
Public view **215**, 238

Q

Query view **260**, 263, 265
Quick sizer **38**

R

Rank node **175**, **176**, 178
Real-time reporting 9, 34, 35, 37, 215, 235
Redundancy **39**, 244, 249
Restricted column 116, 117, 228
Result cache **234**, 235
Reuse view **260**, 264
Row engine 51, 52

S

SAP Data Services **46**, 47
SAP HANA Live **259**, 263, 264
SDA 36, 37, **41**
Semantics node 131, 157

Single version of truth **35**, 244, 247
Sizing 37, 38
SLT 35, 36, 42, 44, 45
SQL engine 52, 140, 143, 222, **237**
SQLScript 39, 43, 51, 164, 208
SRS 42, **45**, 46
Star schema 42, 51, **62**, 220
Statement memory limitation parameter **236**

T

Table type parameter 236

U

Union node **138**
Unique value **70**
Unit conversion 113, 147, 229, 232
Unloading 218

V

Value help view **260**
Variable 91, 99, 130, 133, **156**, 158, 233
Virtual access 41
Virtualization **31**, 35, 39, 40, 48