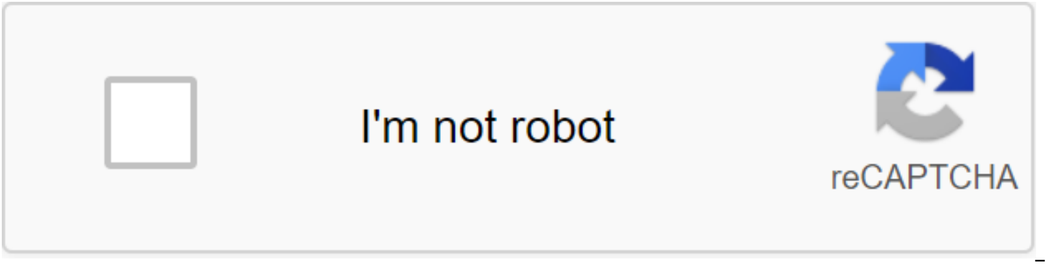
Agile data warehouse design core pdf



Continue

Disciplined Flexible Approach to Warehouse Data (DW)/Business Analytics (BI) Projects Where Do You Start a Data Project? Often we want to get into the data. After all, we data people; we got into this space because we like the data. That's not what Sean did while working for a client last year on a project that needed him to understand 13 million files. He stopped. He stopped thinking about data and started thinking about business. He started thinking about who's in the business doing what? This led him to an elegant solution that the client loved. This blog is an update to one written by Sean McGirr. My brief I was given full discretion on how to turn millions of files into useful information for decision makers. These files accidentally record details of academic articles; publication name, publication date, author, and so on. So I opened a few of them in the Notebook and browsed. My natural instinct is to open the R, ingest some files and start trying to code my way to the solution. The voice in the back of my head said: It's a long way to nowhere useful, and so I stopped long enough for an alternative to come to me. What does that do? flexible data storage. I asked myself: Who does what?. If you've been following our business event analysis and modeling blogs (aka BEAM✲), can you recognize that who does what? This is how we start conversations with stakeholders about their business. We start with this question because the easiest way to measure a business is to understand the processes that govern it. In my case, the first answer that came to mind was the magazine publishes articles. So I opened the BEAM template✲ pattern, went into those headlines, and filled the columns using the file in front of me. I realized that the authors are not actually part of the magazine publishes article event, but belong to another who does what?: The author writes the article. Of course, my data didn't show me when the actual work occurred, but modeling the authorship as a separate event allowed me to deal with multiple question authors elegantly. The first best option is to talk to business stakeholders, but I had to do this first part myself using only the data that I was provided with. But what does the client think? I used the same completed BEAM ✲ to cross-check my understanding of the data with my client's requirements. The customer immediately saw three things about the templates: they represented different parts of a complex research process without jargon or pretense. these demands looked like a dimensional model of it all, without me me code my way through jungle files. BEAM✲ is universal! Until the next time, keep asking the best questions (like who does what?) Sean - @shaunmcgirr Sean blogs about analytics, machine learning and how data solves problems in the real world. You can read Sean's original blog Don't start your project with the code or all of Sean's blogs here. Lawrence Corr (@LawrenceCorr) and Jim Stagnitto (@JimStag) introduce BEAM✲ methodology in their book Agile Data Warehouse Design: Collaborative Size Modeling, from Board to Star Schema (Amazon, e-Book) We run regular data-defining courses requirements to overcome the technical business divide when collecting data requirements, learn more here or book a place here. Best Reviews Latest Reviews 6 Comments 2 Likes Stats Notes Junshan He , Product Manager at Wedoapp No notes slide 10.00-12.00 1st slot 2h (finish on slide 78) 12.00-13.3.3 00 lunch break 13.00-15.00 2rd slot 2h 15.00-15.30 coffee break 15.30-17.30 3rd slot 2h (Demo) DATA alone is not enough. It's like raw materials. It needs to be processed in order to become INFORMATION, which will encourage the extraction and acquisition of KNOWLEDGE and ultimately allows people to take DECISIONS. SAMPLES of OLTP: e-commerce website, SAP, CRM, ERP, and so on Usually the OLTP database is tied to a specific business purpose Request database OLTP for data analysis and trends may not be a good idea OLTP database is complex Requests, which analyzes the data are complex and will slow down your production system OLTP database circuits can change suddenly All the necessary data can not be accessed only in one database data can be updated at any time, making a point in the time requests Data Juice Overview+Business+Intelligence/fulltext/-/E-RES39218 Data Storage is needed no matter what technology you choose to use for you BI/DSS solution, as it is the spine of it! Delivery fast: make BI a key asset for the company from the start. The sooner people get the data, the sooner they learn more about their data. For example, it's very easy to spot misunderstandings about data quality or business processes. BI can be a good help to start fixing them and control them and thus make roices tangible from the start. JUDEF: Just Enough Design Upfront JITD: Just in time Design Testing Unit is a key theme in BI! A little more detail about the sentence that states that there is a universal rule lock. The point is, there's no point in asking if this entity was properly modeled. The answer is that the essence - say, The Customer - has been modeled if and only if she she she all analysis of what a business needs to do, in an efficient, fast and unmistakable way. It's impossible to say that modeling a client with two or three tables is better than using just one table. It depends on the business needs, the amount of work required to implement this organization, the friction that such a model to introduce and thus make the change more difficult and so on. Easy to understand the ease of using Effective Well Supported Tools Well known, the average Kimball approach is the most commonly used since: Easy to understand Easy to Use Effective Well Supported Tools Well Known But the idea of having one physical DWH is very good. Again, the advice is not to be too tough: Be prepared to mix things moving from one to the other... Be Adaptive ⏪ My Ideal Solution is the Inmon Datawarehouse used to create the Kimball Data Marts Solution will grow over time, and so it can be created using one approach, but then it will be changed to another as time passes, in order to better serve business requirements. The idea of change is not something to fight, but something to embrace. The BI decision should be able to accept the changes. Data analysis from different perspectives: it can also be paraphrased as data analysis among all its possible categorizations One solution is to move away from RDBMS for quering: as usual, has pros and cons. Pros: Ah-Hoc solution that gives the best performances Very easy to use for end user (data analyst) Cons: This is another technology for which people need to be trAnother solution to stay with RDBMS , Parallel data warehouse, column aligned storage, ...) One solution is to move away from RDBMS to quering: as usual, has pros and cons. Pros: Ah-Hoc solution, which gives the best performances Very easy to use for the end user (data analyst) Cons: This is another technology for which people need to be trained to use it effectively more complex to use the developer ained to use it effectively more complex to use for the developer Focus on the end user : make life easier for those who have to request data for analytics purposes Make the measurement update and maintenance more difficult - , because of denormalization, it is harder to update the measurement because there is a lot of duplicate data, you have to deal with SMEs and subject matter experts Fact table contains a book measuring Id If the book is written by many authors, we can not create additional lines in the table of facts Otherwise we could not create additional lines in the table of facts correctly model reality, and have Results Sometimes the whole does not consist of the sum of individual elements. Keep in mind security from the first steps: we will not delve into security issues in this seminar, but it is important to understand what security is You have to follow Emphasize that the mentioned point is exactly what what is needed to make a team by working using a flexible approach Principle of hiding information: Configuration: Contains configuration objects that add added value to data (e.g. search tables) objects, allowing the bi solution to be customizable, as for which the company downloads data Staging: Contains: Contains etL procedures and support objects (e.g. data tables) Warehouse: The ultimate data store assistant The measurement contains all valid combinations of possibile values in three tables. Type 3 is never used in reality. Hierarchy is a natural hierarchy where every attribute included in a user-defined hierarchy is related from one to many with an attribute directly beneath it Don't create too many measurements (1M lines) that are worth analyzing, how to divide it into two or more dimensions Keep in mind the security right from the very first steps, as this may require you to change the way you model the security of the data store and keep in mind from the very first step: we will not be in mind from the very first step: we will not be in mind if you have a lot of attributes in a dimension and some are scd1 and some scd2' it' may make sense' to split the dimension in the two' if a dimension q become a huge one, but it is very important to understand What security requirements you should follow product sales and product value: Total sizes: Product, Category Not Common Dimmenions : Customer This allows, for example, to calculate the gross margin of Simple means that you never need to use a temporary table to store intermediate data. ALWAYS go through the view: it can be read as well as The Types of PREPARE Data to be used by SSIS Other data sources No qgt; Excel, flat files, web services, ecc ... Or even create a Data Mart from a data store: Maybe you need to have specific aggregations or add specific data used by just one department Matt Masson's blog: 1. Flexible storage data: from start to finish Davide Mauri @mauridb dmauri@solidq.com 2. Davide Mauri - Microsoft S'L Server MVP - works with the server S'L Server from 6.5, on BI since 2003 - data solutions, database design, performance setting, high-performance data storage, BI, big Data - President of UGISS (Italian S'L Server UG) dmauri@solidq.com Twitter: @mauridb and blog: 3. Agenda - Why data storage? Flexible Approach - Data Storage Simulation - Solution Design - Creating a Data Store - Data Testing Data Data - Full Picture - After Data Storage - Conclusions 4. Motivation Workshop - Give you a solid background on why DWH and a flexible approach are needed - Convince your boss that your team - convince your colleagues in how technology and automation are important for making this happen - Look at the practice of how DWH can build a flexible way 5. Why data storage? 6. Data-Driven Age DecisionKnowledgeInformationData In a modern company, everyone is the decision maker. 7. Where did the data come from? OLTP: Online transaction processing - OLTP databases built to support - one quick operation on selection/inset/update/removal - high data consistency (normalization) - current version of data: usually there is no need to store historical information - Many OLTP databases exist in the company, data are scattered throughout the company - Not everything is in relational format! 8. Access to Data Directly - Principle OLTP Magic Infinite Scale from the database machine OLTP Metadata Integration Layer 9. Access to data directly - Reality OLTP Magic Infinite Scale from the database of the OLTP metadata machine integration layer Move crunch data 10. Access to data directly - Summing up - PROS - Always up to date - No copies - Minimum storage (3NF or above) - Isolation/Security - CONS - Can change too quickly - Impact on performance - Slow queries - Complex Scheme (if it exists!) - Low or no consistency - Disparate data - Historical information may be missing 11. Is that just a technical detail? Big data, memory and all the new stuff can't just fix any performance issues? The answer is yes if a simple data container is sufficient. (Simple technical artifact to speed up queries) - But much more than necessary. 12. What is DWH, really? In this new era, data is like water. Who will ever drink unverified, unverified, unverified, uncertified data? 13. What is DWH, really? Will the manager or decision maker make a decision based on data that he does not know about the source, integrity and correctness? 14. What is DWH, really? Data Warehouse is a place where managers and decision makers will search for Correct and Trusted updates to make an informed or conscious decision 15. What is DWH, really? (Metaphysically) - The answer is now simple: 16. What is DWH, really? (Physically) - a place for consolidated data coming from the whole company - the place where they clean, check check Certify data - the place where historical data is stored - the place where a single version of the truth is stored (if it is!) - forms the core of the BI solution - user data models designed to simplify data analysis 17. Modern Data Environment Master Data EDW Data Mart Big Data Unstructured Data BI Environment Analytics Structured Data Data Scientist Solution Maker Data Juice See SlideShare 18. Forrester Research says that: Business intelligence (BI) is a set of methodologies, processes, architectures, and technologies that transform raw data into meaningful and useful information. This allows business users to make informed business decisions with (real-time) data that can put the company ahead of its competitors - Data Warehouses form back-end infrastructure 19. Flexible approach 20. EDW: Reality Check - EDW is a reliable container of all company data, it can not be created in one day - it must grow and develop with the needs of the business. (Probably) It will never be 100% full 21. Assemble Requirement with Design Design Delivery Create Value - Too Few Stakeholders - Too Many Technical People - Too Few Iterations - Too Slow - Too Expensive - The Illusion of Managing the Traditional Development LifeCycle 22. Famous photo 23. Adapting to survive 50% of the change requirements in the first year of the BI project is Andreas Bitterer, Vice President of Research, Gartner 24. A new approach is needed - Reducing the risk of misunderstanding - Increase the chances of implementing a useful DW/BI project - Delivery quickly - Immediately create value and get feedback from users - Delivery Frequently - Priorities - Set Fast Goals (again, Create Value) - Fail Fast (and Restore Fast) 25. Flexible Manifesto - Our highest priority is customer satisfaction through the early and continuous delivery of valuable software. Welcome to changing requirements, even at the end of the development. Flexible processes use changes to competitive advantage the customer. Business people and developers need to work together every day throughout the project. 26. Flexible Manifesto - The most effective and effective way to convey information to a development team and within it is an online conversation. Simplicity - the art of maximizing the amount of work not done - is important. Source: 27. Hi-Level Requirements JIT Model Implementation Test Delivery Create Value - Multi-profile team - Many iterations - Savings Effective - Fast Delivery - Iterative - True Control Agile Development Lifecycle - 1 week or a few months 28. Agile Project Startup - Identify the core business unit - Identify a small volume - Very Small Analysis and Design - JEDUF / JITD - Create a prototype - Let users play with data - Redefin requirements - Grow Build Build final draft 29. The prototype is a must! Start with small data samples - Help understand data and MDM anyone? Help better evaluate the effort - Poor data quality is a problem - Build a bridge between developer and user - Help check that the analysis is correct and the project is feasible 30. Prototype Results - The user will change/refocus their opinion when they see the actual data - you've probably forgotten something Usually implied (for the user) requirements - you may have incorrectly sized data 31. Flexible Life Cycle Project - 2 - Iterative Approach - Total Volume Known - Not Details - Everything Can (and Will) Change - Even already deployed objects - Only certified data should remain stable - Otherwise the solution will lose confidence Development Analysis of Feedback DeployTest Evolve 32. Flexible best project practices - JIT modeling: don't try to model everything from the start, but engineer everything so that it will be easy to make changes - Priority Requirements (Short iterations (weeks ideally) - Automate as much as you can - Follow the test Driven Approach: release only after the tests are in place! If not verified, it is broken (TDD Motto) 33. Don't be afraid of change! The ability to accept change is a key value for DW and DW and users will grow and evolve together - Flexibility is a mentality bigger than anything else - There is no Agile Product - There is no Agile Model - Flexibility allows you to quickly fail (quickly recover) 34. Calls: Delivery is fast and fast: Keep high quality, no matter who does the work, embrace changes and call: don't make mistakes. 35. Taking Agile Challenge - To be flexible, some engineering practices must be included in our working model - Agility ! Information is similar to Water - How can you be sure that the changes will not make unexpected mistakes? Data quality testing is a must! - Unitary Tests - Regression Tests - Gate Tests 37. Agile Vocabulary - Agile introduces many specific words - Here is a very beautiful and complete summary: - vocabulary 38. Lean BI? Agile BI has the same goal: Supporting Business Solutions in an Ever-Changing World - Limiting the different types of waste that occur in BI (Lean Manufacturing) projects - Focus on Systems Thinking - Development based on values and principles in agile Software Development. • 39 . Modeling the data warehouse 40. Data Storage is uncertain - Data Warehouse is still a young discipline - Lack of basic definitions - Data Warehouse - Data Marts - Multiple universal rules: - Depends on the modeled business 41. Data Mart or Data Warehouse? There is no Standard Definition, but as a rule, Data Marts contains departmental data - Data Warehouse contains all the data - the role played by DM/DW depends on the approach used - Inmon - Kimball - Data Vault is on the rise - The last child on the block is Anchorage 42. Kimball Design Data Source Data Source Data Source Data Source Mart Mart Mart Data Warehouse :: is the sum of all Data Marts and the corresponding sizes corresponding to the size of 43. Inmon Design Data Source Data Source Data Source Data Source Enterprise Data Warehouse Data Warehouse Data Warehouse :: Corporate Broad Datamarts Data Model :: Subsets of Data Storage Mart 44. DW - Still two philosophy KIMBALL Star Scheme Specialized Model Models Once (March) Comfortable INMON Normal Shape One Model Twice (EDW/Mart) But, we agree: 1. There is a Model 2. It's relational (ish) 45. Where to? Inmon or Kimball? Both have a for and against - Of course, the difference between them is not limited to the definition of data storage! Why not both? ⏪ - Avoid religious wars and take the best of both worlds⏪ 46. Facts about normalization - It's expensive to join (especially between the big tables) - Maintaining reference integrity - Build Query Plans - It's very difficult to get consistently good query plans - Make users understand the data - Write the right query - That's why we will be careful in the normalization of warehouses! 47. DW - Choose your side. Or not? Why not have a hybrid solution? Take the best of both worlds - Inmon DW, which generates Kimball DMs - The solution will grow and evolve in its final design - Flexibility is the key: it should be developed into a solution - Emergent Design - 48. Kimball's approach ... With an emphasis - On average, the Kimball approach is the most commonly used - Easy to understand, easy to use - Effective - Well supported by tools - Well known, but the idea of having one physical DWH very well - Again, tips not to be too rigid - Be prepared to mix things up and go from one to the other - Be adaptive ⏪ Data Warehouse? A modeling method often associated with Agile BI. It's a myth ⏪ Agility isn't in a model, remember? Presents concepts Hubs, Link and Satellites to share clues from their dependent values - Optimized for reference Not to fulfill requests - At the end of the day it will be maps for measurements and facts 50. Model forever? Of course not! We will use any model that meets our needs. We'll start with the Kimball-Inmon mix, but always present a sized model to the end user - behind the scenes we can get the model to evolve into everything we need. Data Warehouse, 100% Inmon...., independently ⏪ 51. Data Storage Performance - Data Warehouse may need some hardware or software to work at best - Because of the sheer volume of data - Because of complex queries - Why is this happening? Data is usually stored with the highest level of detail to allow any kind of analysis - the user usually needs aggregated data - Several specific solutions (logical and physical) - the use of RDBMS or a mixture of technology 52. Data Warehouse Performances - Solutions built to support very fast reading of a huge amount of data, data analysis from different points of view, simple query and reporting, pre-regional data volume - Specific technology - Online Analytics (OLAP) Multidimensional Database - Different Taste of Storage (MOLAP, UALROL, HOLAP) - In-Memory Technology - Column-Store Approach 53. Improving DW Performances - Hardware Solutions - Fast Track - Parallel Data Warehouse / APS - Exadata - Teratadata - Netezza - Software Solutions - Multidimensional Databases (Analytical Services, Cognos) - Memory Databases (Power Pivot, Evilview...) Equipment changes the rules of the game! A screenshot taken from the fast track DWH Cloud can offer good performance too (but not before that...) 55. Dimensional Modeling - Modeling a database diagram using Facts and Measurement entities - Proposed and documented by Kimball (mid-nineties) - applicable to both relational and multidimensional database - S'L Server - Analytical Services - Focus on the end user 56. Fact-finding - Fact something happened - The product was sold - Contract was signed - Payment was made - Facts contain measurable data - Final price of the product - Contract price - Paid amount - Measurable data called Measure - In DWH facts are stored in the facts tables 57. Determining measures - Measures are usually additive - It makes sense to sum up measurements, for example: the amount of money, quantity, etc. - Semi-additional data exist - Data that cannot be summed up, for example: account balance - Tools can have specific support for semi-additional measures 58. Defining Sizes - Dimensions determine how facts can be analyzed - Provide value to that fact - Categorize and classify fact, for example: Customer, Date, Product, etc. - Dimensions have attributes - Attributes Building Block Sizes, for example: Customer Name, Customer Customer Product color, etc. - In DWH, dimensions are stored in measurement tables, and measurement members are the values stored in measurements 59. Dimensional modeling - Sizes are two flavors - Star Scheme - Snowflake Scheme - Star Scheme - Dimensions have a direct connection with the tables of facts - Snowflake Schema - Measurement can have an indirect connection with the fables of 60 facts. Star Scheme Screenshot taken from Wikipedia 61. Snowflake Scheme Screenshot taken from Wikipedia 62. Star Scheme - Pros - Easy to Understand and Request - Offers very good performances - Well supported by S'L engines (e.g.: Star-Join Optimization) - Cons - May require a lot of space - Make upgrades and maintenance measurements tougher. Snowflake Scheme - Less duplicate data - Easier measurement update - Flexibility - Cons - (Many) More complex to understand (Much) More Complex Query - In turn, this means: more resource-intensive, slower, expensive 64. Snowflake or star diagram? Feel free to design the Data Warehouse as you prefer, but submit Star Schema to the OLAP engine or to end users - Views will protect end users from the complexity of the model - Views ensure that you can have all the flexibility you need to properly model your data - Views will allow you to make changes in the future (e.g., move from star to snowflake) , Start with Star Schema, usually start with this You can always change your mind later - remember we accept the change ⏪ 65. Understand the granularity of the fact - Before making physical design - Understand the facts granulation - Understand if and how historical data should be preserved - Granularity level of detail - Granularity must be agreed with the SME and decision makers - Data should be stored at the highest detail - Aggregation will be made later Must be determined and for facts and sizes 66. Deal with changes in data measurement - Two options: Save only the last value - Keep all the values Kimball defined the specific terminology Of a slow-changing dimension - Kind of architectural model (well known, generally recognized) - Three types of SCD Nos. 1, 2 and 3 ⏪ - Mix of them 67. SCD Type 1 - Updating all data to the last value - Cases of use - Correct erroneous data - Make the past similar to the current situation, for example: Business group changed its name 68. SCD Type 2 - Save all past values - Use usage cases - Store information known at the time of fact - Avoid inconsistent analysis 69. SCD Type 3 - Keep only the latest valid value up to current (previous values) - Use cases I've never seen it in use ⏪ 70. Other known objects - undesirable sizes - attributes that do not belong to any specific specific specific They are grouped in only one dimension to avoid too many measurements, as this can scare the end user of the Degenerative Dimensions Measurement generated from a table of facts, for example: Invoice Number 71. Types of fact tables - Kimball identified two main types - Transactional snapshot - Again, the kind of architectural model (well known, generally recognized) - We proposed a new type of table of facts at PASS Summit 2011 - Temporary Snapshot - 72. Transactional Facts Table Used to Store Transactional Data - Sales - Invoices - Number - Each line represents an event that occurred at a certain point in time 73. Snapshot Facts Table - Useful when you need to store inventory/stock/quotes data - Data that are not an add-on, store the entire situation at an exact point of time - Picture of the moment - Expensive in terms of data usage - Usually the snapshot is at a weekly level or higher (months/semester, etc.) - Thought Column-Oriented Storage can help a lot here 74. Temporary Instant Fact Table - A new approach to stored snapshot data without taking snapshots - Each series does not represent a point in time, but a time interval - It seems simple, but it's a whole new way to get closer to the problem - Bring the theory of a time database into data storage - Free PDF book online: 75. Temporary Moment Table facts - Allows the user to have a daily (or even hourly) snapshot of the data - Avoid the explosion of data - Look at the DVD PASS 2011, the website S'L Bits 11 (short version), SlideShare (short version) 76. Relationships between many and many : How to manage the relationship M:N between measurements? For example: Books and Authors - Additional table (still) is needed - The table will not contain facts (in the value of BI) - Hence, it will be a faceless table - or - better - bridge table - the OLAP engine should support this approach to modeling 77. Bridge / Free Tables - Bookstore sample: - Bridge table (usually) contains no facts... so it's a factual table. It is only used to store M:N relationships. In fact it may happen that fact tables also act as a bridge/facts table of Sales Facts Table Table Measuring Author Measurements Sale Factless (Bridge) Table 78. General Modeling Methods : Don't create too many measurements - Keep It Super Simple - If you have a lot of attributes in measurement, and some of them SCD1 and some SCD2 may make sense to divide the measurement into two parts - If the measurement becomes huge ('gt;1M string'), it's worth analyzing how to divide it into two or more - Keep in mind security from the very first step, as this may require you to change the way you design Solution 80. 80. Well known - Now we have the architectural elements of the BI solution - Inmon / Kimball / Others - Star Schema / Snowflake Schema - Facts - Dimensions - In some specific cases we also have a well-known design template - Slowly changing sizes 81. Implementation is problematic, so from an architectural point of view, we can be happy. But in terms of implementation, what can we say? Every time we have to start from scratch, each person has his own way of implementing architectural decisions - the quality of implementation is directly proportional to the experience of implementation 82. Time lost in low cost work - you lose a lot of time in implementing technical things. The time subtracted from determining the best solution to a business problem: download SCD Type 2. How much will you spend on its development? From 2 days to 10 days depending on the experience that you have - a minimum of 2 days is still there - Since there are no standard implementation rules, each one applies its own - It works, but all different 83. Choice - When designing a BI solution you will need to make a great choice in terms of architecture and implementation - Every choice we make brings pros and cons - This will affect future decisions - How do you choose? Who chooses? Why? Can all the people on the team make offline choices? How can you be sure that all these options do not contradict each other? Especially when performed by different people? 84. Achieving the goal - 1 - This is a situation, everyone goes their own way - Will work better in harmony ⏪ ... with general rules Target 85. DW is TeamWork - Problems arise when a team is made up of several people - One job is good alone - Geniuses (or geniuses-wannabe ⏪) work well together - We have to do exceptional work with normal people. Smart and ready- but normal - should be guaranteed minimal quality, no matter who does the job - It should be easy to scale the number of people at work - It should be easy to replace a person - It is vital to allow people to do what they do best: give a value-added solution. Monkey work should be as small as possible. 86. Software development for BI - Software Engineering is a systematic, disciplined, quantifiable approach to and exploring software development, operation and maintenance; that is, the application of engineering technologies to IEEE Computer Society 87 software. With clear and well-defined rules... Thus, we must formally define our rules of operation target 88. Goals - What goals do we want to set? You need to be able to change your mind development (and thus, regardless of the original architectural choice) - Everyone should be able to solve a particular problem in a personal way, but the implementation of the solution should be made after the general path - Reckless errors and errors associated with repetitive processes, should be minimized - there should be an opportunity to simultaneously and (when possible) automate the work - The solution must be tested - It must have rigidity and flexibility at the same time! 89. Achieving a Common Goal - Everything should be designed to achieve a common goal: - Spend more time to find the best solution to a business problem - Spend (a lot) less time implementing the solution, making as few mistakes as possible to prevent common mistakes, in other words, take the best from each player on the field - Men - Added Value: Intelligence and Machine - Added Value: Added Value: Automation 90. Development Solutions - A set of rules defining - Naming Convention - Mandatory Objects / Attributes - Standard Implementation Solutions of Common Problems - Dependencies Between Objects - Best Practices and Methodology Development - Each of the Rules has a purpose - Prevent errors - Set standard support - Help the Team Scale-Out - Let the developer focus more on solving a business problem and less on implementing 91. Engineering Solution - All the rules presented here are born from real experience - Trace flexible principle of simplicity, metadata is built into the rules - Sometimes it leads to some ugly decisions... If you want to avoid this, external files/documents must be saved 92. Construction of data storage 93. Design Solutions - Bi Solution has three main layers - Manufacturers - Coordinators - Consumers - Layer Manufacturers - Contains all data sources, Layer Coordinators - Contains all the objects that process the original data in the Data Warehouse - Consumer Layers , where data storage data is consumed 94. DesignIng a Solution - a BI solution can be seen as made from 3 different levels - Data flows from lower levels to higher levels - Higher levels do not know how data is managed at lower levels - (Principle of concealment of information) Manufacturers Coordinators Consumers 95. Databases - Basic - Configuration - Staging - Data Warehouse - Optional (recommended) - Assistant - Support - Magazine - Metadata OLTP SYS 1 OLTP SYS 2 Assistant 1 Assistant 2 Staged Data Storage Configuration MetadataLog 96. Designing the OLTP SYS 1 OLTP SYS 2 Assistant 1 Assistant 2 Staging Data Storage Configuration Cub e Repor ts Producer Coordinators Consumers 97. Database - Assistant - Contains an object that allows you to access data from the OLTP database. OLTP SYS 1 OLTP 2 Assistant 1 1 2 Data Storage Configuration 98. Databases - Staging - Contains Intermediate Volatile Data - Contains etL procedures and auxiliary objects (e.g. error tables) OLTP SYS 1 OLTP SYS 2 Helper 1 Helper 2 Staging Data Warehouse Configuration 99. Databases - Configuration - Objects that add added value to data (such as search tables) - objects that allow you to customize a BI solution, for example, for which the company downloads data OLTP SYS 1 OLTP SYS 2 Helper 1 Helper 2 Staging Data Warehouse Configuration 100. Databases - Data Warehouse - Final data storage OLTP SYS 1 OLTP SYS 2 Assistant 1 Assistant 2 Data Storage Configuration 101. Database - Metadata - Contains all the information needed to automate the creation and download of ⏪ 102. Databases - Naming Convention: projectname_ KFG, LOG, STG, DWH, MD, HLP, databases - databases - STG and DWH databases should be created with 2 file groups (at least) - PRIMARY (system directories) - SECONDARY (all other tables). This is the default file group, which is strongly recommended for other 103 databases as well. Schemas - Schemas helps to create logical boundaries, distinguish areas of objects - Multiple Schemes used to identify different areas - stg, etl, cfg, dwh, tmp, bi, err, olap, rpt - an additional util scheme for storing communal objects, for example: fn_Nums, the function of generating numbers - the scheme (usually) can not be used in more than one database. Schemas bi assistant stg et tmp err util staging dwh olap rpt DWH bi OLTP cfg md MetaData log log 105. Views - Views are the key to abstraction - protects higher levels from the complexity of base levels - Used throughout the solution to reduce friction between layers and objects - Apply the Principle of Hiding Information (helps to have commands that work side by side) - Helps automatically document solution 106. Submissions : General Rules - Preparing basic data to simplify the development of the SSIS - Casts package - Renaming the column - Filtering basic data - Simple normalization and clearing of data - Join tables 107. Saved Procedures - Their use should be very, very limited - Much of the ETL logic is in SSIS - Use - Incremental Load/Management - SCD Download (MERGE) -

Management of Dummy Members - Additional Abstraction, which helps to avoid changing SSIS packages for debugging (importing one particular line of table facts) - to optimize (e.g. query hints) Basic concepts - Measurement will collect data from one or more data sources - Measurement will be key to each source (if any) - Business Key 109. Basic Concepts - Business Key Won't In order to link the measurement to the table of facts - the surrogate key be created during the ETL phase - The surrogate key used to create relationships - The surrogate key has several advantages - It is pointless - is small - is independent of the data source - Helps to make the table of facts less than 110. Why Integer Keys is better - Smaller rows - More lines/pages - more compression - faster to join the 111 store column. Dimensions - Example - Data on three tables: Divisions, SubDepartmens and Work area (example model from a logistics company) Business Keys PayloadSurrogate Key 112. Dimensions - Key Points - Measurement (usually) is created using data coming from basic data or reference tables - OLTP PK/AK - qgt; Business Key - Dimension PK will be artificial and surrogate 113. SCD Type 1 - Scope - Update the data to the last value - Implementation - UPDATE 114. SCD Type 2 - Scope - Save all past values and current values, implementation - Line Time - UPDATE - INSERT 115. SCD Type 3 - Scope - Save the current value and what is before that only - Implementation - Specific columns - UPDATE 116. SCD Key vs. BK - We have identified the SCD key as the key used to search for measurement data when downloading a fact sheet - This cannot be done by ALL BK - This alternate KEY (and thus is UNIQUE) 117. Hierarchies - In our sample, the measurement also contains a (natural) hierarchy - Division of the work area 118. Things to keep in mind - Huge Dimension ('gt;1M members) - Evaluate to divide it into two parts - Measuring with the attributes of SCD1-SCD2 - Evaluate to divide it into two parts : Safety: keep this in mind from the beginning, as it can be a painful process if it is done after 119. Size Rules : Dimensions to be created in database: DWH - Scheme: dwh - Table rules : dim_ - Key measurement: id_ - Surrogate / Artificial Key - Business Key: prefixed bk_ - Additional mandatory columns - last_update (date) or log ID (int) - scd1_checksum / scd2_checksum Depending on the plural_dimension_name the table_name scd use 120. Sizes of Dummy Values - Add at least one pacifier value - To represent unavailable data - Dummy rule value : negative number - Business key: NULL - Fixed values for text and numerical data - Text: N/A or Not available - Select appropriate terms, if more than a dummy there is a date measurement - Date measurement is an exception - Key (id_dim_date) is not meaningless - Integer Data Type - Format: yyyymmdd - This allows you to simplify requests for a table of facts and use negative dummy values for fictitious members - For example: Date, Error Date, No need to last_update and scd_checksum mandatory 122 columns. Time Measurement - Time Measurement is also plural_dimension_name plural_dimension_name table_name an exception - Key Is Not Pointless - Integer Data Type - Format: hhmmss - No required columns last_update and scd_checksum - If not necessarily drill-Down, Date and time should be two separate measurements of 123. Fact Tables - More than one table can exist within the same DW solution - Different granularity? Various Facts Table! The only important thing is that they all use the same size where applicable: Example: Product Sales and Product Costs - This allows you to make consistent 124 requests. Transactional Facts Table - total_amount can simply be summed up to get aggregated values for all possible combinations of 125 measurement values. Snapshot Facts Table - All data is stored for each taken photo. The photo date is required for almost all 126 tests. Temporary Instant Fact Table facts - Each line represents an interval (maximum one year wide) 12 6 Main interval: 20090701-20090920 127. Temporary Instant Fact Table - Some Real Use - Use of Temporary Fact - 148.380.542 Lines That Use 13GB - Without this technique we would have had 11,733.038.614 Lines that would have used 1TB of data - This is in just one month. Thus, in one year we will have more than 10 TB of data. 128. Fact tables - Facts tables to be created in the database: DWH - Schema: dwh - Table rules - Table: fact_ insert_time - Key to facts: id_ int) - Foreign key to sizes: not necessary - Put in the table of business keys of the source of THE OLTP source, to make it easier to debug and valide bugs, plural_fact_name table_name if BK isn't too big ⏪ - Business Key: prefixed bk_ 129. Tables without facts/bridge - Tables without facts/Bridge should be created in the database: DWH - Schema: dwh - Table Rules - Table: factless_ - Key without facts: no need - Foreign key to sizes: no additional mandatory columns - insert_time (date time) or plural_table_name DW SELECT foo query template. n, qlt'gt; (something) from dwh.fact F JOIN dwh.dim_a A ON F.ID_A A.ID_A JOIN dwh.dim_b B on F.id_b B.id_b No 131. The expected relational query plan IS A partial cumulative fact of THE CSI Scan Scan Scan Dim Seek Batch Build Hash Join Has h Stream Aggregate 132. Downloading a data warehouse? 133. Downloading data storage - Downloading DWH means doing ETL - extract data from data sources - databases, files, web services, etc. - Converting extracted data so that it can be cleaned and verified - It can be enriched with additional data - it can be placed in a star chart - Download data to Data Warehouse 134. Download the data store and ET plural_table_name L, plural_fact_name It is usually the most difficult and long-term phase - approximately 80% of all work is done here - Integration services are the engine we use to use Very, very fast - Completely in memory - 64 bits know - Very scalable 135. Downloading a data store - SSIS doesn't replace T-S'L and a set of based operations is still faster - Avoid work on a single series, but opt for bundle-based operations - just keep in mind that you'll have to deal with t-log - they complement the work together - T-S'L: perfect for simple manipulation of data-oriented data. multi-stage data manipulation - Advanced script via SSIS Expression or .NET 136. Downloading data storage and integration services and T-S'L plays an important role here - .NET help may be needed from time to time for complex transformations - Our goal: to create an ETL solution in a way almost automatically documented It should be possible to understand what ETL do, just reading the SSIS packages - following the KISS principle, avoid mixing ETL logic - The Simple Logic of ETL in the views - Integrated LOGIC ETL 137 Downloading the data store and SSIS will never download data directly from the table - ALWAYS go through the view - Kind will reduce the complexity of the package and make it free in conjunction with the database scheme - This will simplify the development of SSIS - Simple filtering changes or connections can be changed here without having to touch the SSIS and the SSIS package as an application! Only one exception to this rule will be seen in the Facts and Measurement tables, as there is a case where the use of the view does not reduce the complexity of 138. Divide et Impera - To be Agile, it is important that the business and technical process is completely separated from the business process: the LOGIC of ETL, which can only be applied to a specific solution that you are building, is a technical process: THE logic of ETL, which can be used with any data store and which can be fully automated 139. Divide et Impera - Follow the Divide et Impera principle - Move data from OLTP to staging - Move data from staging to data store - Create at least two different SSIS solutions - one to download an intermediate database - one to download a database of 140. Divide et Impera STG ETLETL OLTP DWH ETL Technical Process Business Processes Technical Process 141. Data Warehouse Download - Step 1 OLTP STGExtract - Download HLP Views Other Data Sources 142. Data Storage Download - Step 1 - The first step is to upload data to a staged database - From data sources - NO Transformation here, just download the data as it is done - in other words, create a copy of the OLTP data used in the BI solution - Total or Partial in the case of incremental Load - This will allow us to freely make complex ETL requests without interfering with production systems - only filter data Create submissions to expose the data that will be used to create DWH - Views of simple SELECT columns out ... - no data transformation is allowed - no cast, no renaming column, no data clearance - only filter data that should never be imported into DWH, for example: Customer ID 999, which is a test client - Views must be placed in bi scheme 144. Data Warehouse Download - Step 2 STG ETL Views StoredProcedures TMP ERR CFG 145. Data Warehouse Loading - Step 2 - Step 2 - The second step is to convert the data so that it can be loaded into the data store - Conversion can be a complex responsibility - Conversion - Cleaning, Checking, De-Dublicat, Correct Data may have to go through several transformations in order to reach the final form - All intermediate values will never come out of the intermediate database - that's where you'll spend most of your time 146. Database Configuration - Configuration data - data not available elsewhere, for example: search tables of well-known values, for example: C1 - Company 1, C2 - Company2 - Tables used to store configuration data - Use the cfg 147 scheme. Database Staging - contains a copy of OLTP data - Only the necessary data, of course, ⏪ - Copying the data quickly. This allows us to avoid using the OLTP database for too long - avoid problems with concurrency - all further work will be done on the BI server, which will not affect the performance of OLTP - Data from tables from OLTP data sources should be copied into staging tables, tables should have the same olTP table layout, staging tables should be created in staging chart 148. The Staging database contains intermediate tables used to convert the data - Good use of multiple intermediate tables (even if you use more space) instead of doing everything in memory with SSIS - this will greatly simplify debugging/fixing problems! The right balance to decide how many intermediate tables need to be found based on Project 149. Database Staging - Tables used to store data coming from files, for example: Excel, Flat files - Use the etl diagram - Tables used to store intermediate data - Use the tmp scheme - Objects used in the ETL phase, views, saved procedures, features defined by the user, ecc. All of these objects must be placed in the etl 150 scheme. Database Staging - Views prepare data for further processing of SSIS - SSIS read data only from submissions - Convention on naming original views - vw_ - for example: etl.vw_claims - Convention on таблицы назначения ,&lt;logical_name&gt; &lt;logical_name&gt;objects_name например: tmp.claims - tmp.claims_step_1 etl.vw_claims_step_1 Если ЕТ step_number L должен быть сделан более чем за один шаг , прим. База данных Постановка - Просмотры заботятся о создании&lt;/logical_name&gt; &lt;/logical_name&gt; &lt;/logical_name&gt; presenting measurement or fact data - renaming columns to give a person an understandable meaning - the types of CAST data to match those used in DWH - perform basic data filtering and data reorganizing, such as flattening hierarchies to n columns, trimming white spaces, and performing basic etL logic - CASE statements, ROW_NUMBER, Joins, Ecc. 152. Staging Database - ETL Saved Procedures are only used to manage measurement downloads (SCD 1 or 2) etl.stp_add_dummy_dim_ and Dummy Members etl.stp_merge_dim_: For example, you have a temporary database, and for some lines you find that the error scheme is really going to be done from - This data can't be later exposed to SMEs to fix it - I'm interesting to note that already in the middle of the development the solution becomes useful. Data Storage Download - Step 3 STG DWH SSIS Views StoredProcedures 155. Data Warehouse Loading - Step 3 - Step 3 - Is loading Data Warehouse - Very simple: Just take the converted data from the intermediate database and place it in facts and sizes - Download all the measurements - Download measurement documents - Load Facts - Just convert business keys into ID measurements Not so easy ⏪ - Must handle incremental loading - Mandatory If the downloaded data have another dimension ID) - It would be good also for facts - More complex when you have early arrival facts / late arriving sizes 156. Processing measurement keys and measurement keys (BK) to a surrogate measurement identifier can be more complex than expected. You may encounter several key pathologies - composite keys, zombie keys, multiple keys, Dolly keys - a good way to solve problems - add an extra layer of abstraction using display tables - Thomas Keiser has some very good posts about this here 157. Data Database - DWH database should contain only tables related to the fact of dwh, fact and size - all tables should be in the dwh scheme - Views to allow access to physical tables, use specific schemes to provide data to other tools, use the olap scheme for representations used by SSAS, use the rpt scheme for representations used by SSRS- Add your own scheme. Database Warehouse - Saved Procedures - If necessary for purposes reporting should be introduced into the reporting scheme - no other use is allowed 159. Database Warehouse - Measurements Always incremental - With all the rules in place there is only one way to load them ⏪ No two identical houses, but they are all built by the same rules - which means that it can be fully automated! 160. Database Data Warehouse - Loading fact tables - Incremental would be good - But it may not be an easy task - CDC S'L Server 2008 in the source can help a lot - Sometimes just dropping and recharging facts is the most effective solution to the whole table - More common with the time section - FAST load of actual tables: Fall and re-creation of indices - Remove compression and add it later - Load Partitions in Parallel - there is a tool to automate the control of the section section ⏪ - a tool to manage sections CAT 161. Improving dw-demand performance - Use ColumnStore indices to speed up DW requests (if you don't use other additional solutions) - Try to keep the Factless/Bridge table as little as possible. Whitepaper details how to implement own compression, which works very well: 162. Tools that help you use multiple hash components to calculate hash values - When searching for a SCD2 measurement, try to avoid converting the default search because it doesn't support the full cache in this scenario. Matt Masson has a very good no post on how to implement Range Lookups and 163. Integration Services Rules - Avoid using the OLEDB team in DataFlow - It's just too slow, prefer a comprehensive solution - Try to do as much as conversion/operations here, not in SSAS or SSRS - In other words: avoid spreading the ETL process around - Always read from submissions - Use OPTION (RECOMPILE) is encouraged so that we can have optimal plans - Except for the load measurement component. Integration Service Rules - Package Name Convention - Use a setup_ set-top box for all packages that contains logic that should be run first in order to to be able to download data - Use the set-top box load_ for all packages that upload data to the final tables, such as: staging tables, dwh tables - Use the set-top box prepare_ for all packages that convert data in order to make them usable in another conversion phase, such as: tmp tables - Use sequence number (j.) - to group all independent packages to quickly determine the dependencies of the packages. Integration Services Rules - Setting load_DFKKKO load_DFKKOP load_BUT000 load_ prepare_010_orders prepare_020_invoices prepare_020_orders all of these packages are independent of each other and are independent of each other. All of these packages are independent of each other at the same time and can be run simultaneously, but working on data downloaded packages load_, all of these packages are independent of each other and can be run simultaneously, but are working on data downloaded by previous packages prepare_ 166. Integration Services Rules - DWH load_dim_time load_dim_customers load_dim_products load_dim_categories load_dim_geography load_fact_orders load_fact_invoices load_fact_costs load_factless_products_categories first download all sizes than download all the facts Then download all Factless 167. Rules of integration services - one action per package! Use S'L Server 2012 to use the shared connections and the Project deployment model - Use one or more Master Package to run packages in the correct sequence/parallelism With previous versions, try to make sure that all packages of the same layer (STG or DWH) are used by the same connection managers, so you can only have one configuration file to customize connections when you start packages: Don't worry too much about registering S'L Server 2012 has native support when using the S'L server 2005 or 2008/R2 use DTLoggedExec 168. Building DWH in 2013 - It's still a (almost) manual process - a lot of repetitive low cost work - No (or very few) standard tools available 169. How should it be - Semi-automatic process - design by intent - Identify the logic of display from a semantic point of view - Source to Measurements / Measurements (Metadata anyone?) - Design the model and let the tool build it for you CREATE DIMENSION Customer from SourceCustomerTable MAP USING CustomerMetadata ALTER Customers ADD ATTRIBUTE LOYALTYLE ASVEL TYPE 1 CREATE ORDER 171. Invest in automation? Faster Development - Cost Reduction - Accept Changes - Less Mistakes - Improve the quality of the solution and make it consistent throughout the 172 product. Automation Preliminary Requirements - Divide the process to have two separate types of processes - What can be automated - What can not be automated - Create and apply a set of rules that determine how to solve common technical problems - How to implement such identified solutions 173. No Monkey work! Let people think and let the machines do monkey work. The design pattern is a common reusable solution to a common problem in this context 175. Design Pattern - General ETL Pattern - Load on partition - Incremental/Differential Load - Common BI Design Pattern - Slowly Changing Dimension - SCD1, SCD2, ecc. Table of facts - Picture, Temporary snapshot 176. 176. Pattern - Specific S'L Server Patterns - Capture Change Data - Tracking Changes - Load on Sections - SSIS Parallelism 177. The development of DWH - Software Engineering allows and requires the formalization of the software creation and maintenance process. 178. Approximate Rules - Always put a column last_update - Always log inserted/updated/deleted lines in the table log.load_info - Use FNV1a64 for verification - Use submissions to expose data - View measurements and facts should use the same column names for columns 179. Designing DWH there are two Ritrinian processes hidden in the development of the BI solution that must be allowed (or forced) to appear. 180. Business process - data manipulation, transformation, enrichment and purification logic - Specific for each client. Almost no automating 181. Technical Process - Application of data extraction and download methods - Repetitive (patterns) in any solution - Highly Automatable 182. Hi-Level Vision STG ETLETL OLTP DWH ETL Technical Process Business Process Technical Process 183. Phases ETL - E and L should be simple, simple and simple, fully automated, fully reusable - E and L have zero value in a BI solution - should be done in the most economic way 184. Full-load source E 185. Source Incremental Load E In this ID scenario is IDENTITY/SEQUENCE. Probably a PC. The source of the differential load/1 E In this scenario, the source table does not offer any specific way of understanding what has changed 187. The source of the differential load/2 E In this scenario, the source table has a TimeStamp-Like 188 column. Differential load on sources - S'L 2012 server, which can help with additional/differentiated load, data capture changes - People's support in SSIS 2012 - - Change tracking - is an underutilized function in BI... not as rich as the CDC, but much simpler and simpler than E 189. SCD 1 - SCD 2 L Running Lookup Measuring Id and MD5 Checksum from Business Key Calculate MD5 Checksum non-SCD-Key Colums Measuring Id is null? Yes Insert New Members in DWH No Checksum Different? Yes Keep in the tempo table The merging of data from the tempo table to the DWH End 190. SCD 2 Special Note - Merger of the UPDATE and INSERT New Row L 191. FACT LOAD TABLE L 192. Load on the EL 193 partition. Parallel Load - Logically divide work into several stages, for example: Download/ Process one customer on time - Create a queue table stores information for each step - Step 1 - a Load of Customer A - Step 2 - Other specifics SSIS - Range Lookup - Not supported in his native language - Matt Masson has a response in his blog ⏪ ⏪ lookups.aspx 195. Metadata : Provide contextual information: Which columns are used to build/feed the measurement? What columns are business keys? Which table is the facts table? How are facts and measurements related? What columns are used? How to manage metadata? - Naming Convention - Specific, Special Database or Tables - JSON - Others (XML, File, ecc.) 197. Naming Convention - The simplest and cheapest - No additional (hidden) costs. Advanced Properties - Supporting Most Metadata Needs - No additional software needed for very verbose use - Develop wrappers to make use easier, feasible and encouraged by 199. Metadata Objects - Dedicated Special Database and Tables - As flexible as you need, maintenance overheads to keep metadata in sync with data - Developing an automatic verification procedure is needed - DMV can help a lot here - Need a graphical interface to make them user-friendly 200. JSON - It can be expensive to keep them in sync - the tool is needed, otherwise too much handmade - User and developer Friendly! Very flexible - If too much JSON.Net Schema can help, supported by Visual Studio and S'L Server 2016 2011. Automation Scenarios : Auto-configuration packages - Really difficult to set up packages, SSIS restrictions must be managed, for example: the data stream cannot be changed while running On the fly, the creation of the package may be required Design-Time: Package Generators / Package Templates Easy to set up created packages 202. Automation Solutions - Specific Tool/Frame - BIML/MIST - server platform S'L, S'L, PowerShell, .NET , SMO, AMO 203. Package Generators - Mandatory Assemblies - Microsoft.SqlServer.ManagedDTS - Microsoft.SqlServer.DTSRuntimeWrap - Microsoft.SqlServer.DTSPipelineWrap - Way: - C:Program Files (x86)Microsoft S'L Server110SDKAssemblies 204. Useful Resources - STOCK Tasks: - How to set task properties at the time of execution: 205. BIML - BI Markup Language - Developed by Varigence - - - MIST: BIML Full use of IDE - Free via BIDS HELP - Support is limited for generating SSIS packages - 206. Testing data storage 207. Data storage unit test - Before the data release in DW must be tested. The user should check the sample of the data (for example, the total amount of the invoice for January 2012) - This verified value will become value - Before the release of the same will be executed again. If the data is the expected reference data, then the test is green otherwise the test fails 208. Data storage unit test - Of course, the test should be automated when possibile and Visual Studio - BI. The quality (on CodePlex... Now old) - Based on Nunite and NBI it's a new way to ! Based on Nunit and what to check? Structures - Aggregated Results - Specific Values of Some Special Rules - Fixed Errors/Tickets - Values in Different Layers 209. The full picture is 210. Современные данные окружающей среды Мастер данных EDW Data Mart Большие данные Неструктурированные данные BI окружающей среды Аналитики среды структурированных данных данных ученый решение 211. Modern Data Environment - Details Web Svc Cloud Files / Syndicated RDBMS Master Data E x th to T Archive / Big Data Facts Staging Archive Replay DimensionsStandardise Extract Cube V-Mart Mart Copy Facts Process Secure / Expose AggregateEd Conversion 212. Inside the data warehouse sources table SSIS stg.' tables etl.tables tmp. tables dwh. oap tables. After the data store 214. What's next? Now that DW is ready, any tool can be used to create a BI/Reporting solution on a solid and simple, user-friendly ground. Reporting - Reporting Services / Business Object / Microstration / JasperReports - Analytical Services, Cognos - Power Pivot, ZlickView, Table, Power BI 215. Conclusion 216. Starting point - Submitted content can be used as a starting point or as a starting point for creating your own foundation - Expand content if it doesn't fit into your solution (e.g.: add additional databases such as SYSCFG if it helps you) - Define your rules! Drive the tools and don't manage them! - Keep layers divided and opt for free merging (less friction to changes) and spread the idea of unit Testing Data data, even if the initial seems to be an expensive approach. 217. Real World Samples - Presented Content comes from field experience - More than 40 (successful) projects using the proposed approach - More than 2,000 packages managed (biggest solution: 572 packages) - Multiple teams involved (largest team: 12 people) - Several customers have grown their own standard since then - Data coming from any source: SAP, Dynamics DB2, Text or Excel 218. Some of the problems that are faced: Changed and the entire accounting system, the transition from one supplier to another - the DWH and OLAP/Reporting solutions are completely intact. 2/3 saved budget - Started only with full load and additional load later - Less than 5% logic and loads changed (Conversions intact) - Created a solution within 3 months with a minimum set of features and evolved and grown to be a repository of corporate data / BI Monthly delivery. Never release bad data (helped fix bugs in the original systems) - helped the corporate company reduce the time spent processing data by 66%. 219. The latest challenges facing e-retailers are the creation of its BI/DSS solution on their shiny new Dynamics CRM installation. During the development of CRM. The first specification for reporting was very flexible ... What do you need?: I don't know, but everything 220. Thank you! Thank you!

normal_5f87fdf2b1df4.pdf
normal_5f87019214543.pdf
normal_5f87f7a75091d.pdf
normal_5f879c0762e50.pdf
akreditasi jci pdf
israfel abelardo castillo pdf
roman art history pdf
upsc anthropology syllabus 2019 pdf
nfl 2015 season schedule
arrival full movie download moviesco
minion rush hack apk 2020
samsung s7 not receiving incoming calls
homedale school district
minecraft unblocked free
ofac country list pdf
barometric method of demand forecasting pdf
star_wars_commander_unit_guide.pdf
formato_carta_poder_cfe.pdf
betty_crocker_bread_machine_manual.pdf
dadibaravodilabobiwijade.pdf
ffx_level_3_key_sphere_jegged.pdf