

Power BI and SSAS Tabular Interview Template

Data Modeling

1. What are the top goals and priorities of OLAP/Analytical Data Models?

- Provide performant and scalable data source for BI and Reporting Tools
- Provide user friendly, intuitive source for self-service browsing and report development
- Provide single version of truth; apply business definitions/logic (Net Sales = X + Y - Z)
- Protect and secure sensitive data; ensure users only see relevant, appropriate data

2. What are some design practices or techniques you would look to implement in the model to deliver fast query performance and scalability?

- Provide minimum cardinality required for columns (1.23 instead of 1.234235)
- Avoid querying and loading columns and tables not needed by reporting and analysis
- Denormalize model – limit or reduce the # and cardinality of relationships in a model
- Determine optimal sort order to compress key columns
- Use partitions to more easily manage and process/refresh large tables
- Store minimum history required
- Identify most important columns and queries for targeted optimization
- Optimize most important DAX metrics and queries to leverage storage engine, parallelization

3. What is the difference between the Storage Engine and the Formula Engine in Power BI and SSAS Tabular?

a. How do they work together and why is it important for performance?

- Formula Engine produces query plans and can handle complex logic/functions but is single threaded and doesn't have a cache
- Storage Engine receives query plans, is multi-threaded, has a cache, but can only implement simple logic such as joins between model tables, basic aggregations, simple filter conditions

4. What are some design practices you would look to implement to deliver a user-friendly, intuitive and robust experience with the model?

- Apply User Friendly column names/aliases
- Hide columns and tables not needed by users or report developers
- Create hierarchies to provide parent-child drill down capabilities
- Use perspectives to only expose data relevant to a given role (rather than entire model)
- Denormalize the model to reduce the # of tables; follow star schema
- Use display folders (possibly with subfolders) to enable browsing metrics and/or columns
- Apply formatting to metrics and columns
- Define Sorting Order if applicable
- Define Data Categories (e.g. zip code) and set or remove Default Summarization

5. Can you describe the differences between database schemas (Normalized (3NF), Snowflake, Star?)

- Why would one schema be preferable to another for business intelligence/OLAP models?

- Normalized (3NF) is optimized for highly selective writes in OLTP applications
- Star Schema: optimized for aggregations of many rows; simplified & de-normalized tables
 - Single relationship or 'hop' between fact and dim tables
 - Ideal for DW/BI models
- Snowflake Schema: Like star schema but additional relationship(s) between dimension
 - Product category dimension related to Product dimension

6. Can you describe the differences between fact tables and dimension tables?

- What types of columns would you store on each?
- Why or why not should fact tables be related/joined to other fact tables?
- Why or why not should dimension tables be related/joined to other dimension tables?

- Fact Tables store measurements of business process activities and foreign keys representing dimensions
- Dimensions represent business entities that generate and interact with business processes
- Fact tables should not be joined directly to other fact tables given the performance impact of multiple relationships being used at query time (Dim to Fact A , Fact A to Fact B).
 - Additional table also complicates the model, impacts user experience
- Dim tables should not be joined directly, generally, to provide simplified user experience
 - Also avoids scenario where filter applied to child dimension doesn't flow to parent dimension

7. Can you describe the logical process for designing an analytical/OLAP data model?

- What steps would you follow in producing a blueprint or logical diagram of what would later be physically implemented?

- Identify the business process or fact table to be measured
- Determine the granularity of this table.
 - Ensure that each row has the same meaning/definition
 - Determine volume of history needed (2 years, 5 years?)
 - Determine refresh frequency needed (nightly, hourly, real time?)
- Identify the dimensions related to the fact table/business process
- Identify the measures to be created
- Determine if multiple fact tables (Budget, Actual) will be needed

8. What are the differences between models that import data and Direct Query models?

- What advantages and limitations are there to Direct Query or Live Connection Models?
- What advantages and limitations are there to Imported mode?

- Direct Query Models do not store the data, they translate client MDX/DAX queries into SQL queries and pass this to the source system and return results to the client application
 - Pros: Real-time access to source system and avoids moving/scheduling refresh
 - Cons: Limited # of data sources, only 1 data source per model, limitations on DAX functions, performance depends on source system and database
- Import Models physically move/load data from source systems into cube (local model or PBIX)
 - Pros: In-Memory, Compressed Columnar structure for high performance, import from multiple data sources, Full DAX support
 - Cons: Scheduled refresh/import required, hardware required to store model (enough RAM, fast CPU), 1GB compressed limit in PBIX/Excel

9. In import mode models, how is the data stored?

- a. What are the implications of this in terms of model and metric/query design?
- b. What is the primary cause of one table or column consuming more resources than another?

- Data is stored in row segments (8M/segment by default), in memory, in columnar format, compressed based on unique values
 - Thus, the model should look to reduce memory footprint, particularly of columns that are scanned frequently and improve compression.
 - Queries should generally reference individual columns and minimize the # of scans of expensive columns
- Cardinality is the primary cause of a column to consume more memory and thus be slower to query. This is much more impactful than data type.

10. What are some of the primary reasons for poor overall performance?

- Infrastructure:
 - Out of RAM, Paging to Disk
 - Slow RAM (DDR3), slow CPU (2.2 GHz) cores, few cores
 - Single Query server supporting many concurrent users
- Model
 - Snowflake or more relationships;
 - Unnecessary tables or columns
 - Fact to fact relationships
 - Calculated columns
 - Greater cardinality than necessary
- Query and Metric Syntax:
 - Improper use of FILTER
 - Applying filter conditions from the outside in rather than the inside out
 - Try to pass the most selective filter condition via CALCULATE or CALCULATETABLE such that subsequent operations

11. What is a conformed dimension?

- A conformed dimension is a dimension that can be used by multiple fact tables and models thus providing A) single, common (intuitive) version and B) increased productivity in model design

12. Can you explain the difference between natural keys and surrogate keys?

- a. How would they be used in your data model?

- A natural key or business key identifies a unique dimension member in a given source system (e.g. customer # ABC123)
- A surrogate key is a unique, identifying column applied in the BI/DW system (outside of source applications)
 - Surrogate keys are important when integrating source systems and providing support for slowly changing dimensions

13. Can you describe different types of slowly changing dimensions (Type 0, 1, and 2?) and give an example of a dimension that implements Type 2 SCD?

- Type 0 – Dimension Row/Member Never Changes
- Type 1 – Dimension Row/Member is Updated to latest/current state
- Type 2 – New Row is inserted reflecting latest/current state

Some dimension columns may follow different SCD rules/requirements

Type 2 Example: Frequent customer changed addresses and a new row with new surrogate key (same natural key) was inserted reflecting new address. (Historical reporting not impacted)

14. What numeric data types are available in Power BI and SSAS Tabular?

- a. What are the considerations in using one data type versus another?
- b. What date data types are available in Power BI and SSAS Tabular? (considerations)

- Whole Number, Fixed Decimal Number, Decimal Number
 - Whole Number and Fixed Decimal Number (scale of 4) avoid issues with rounding and provide better performance (Whole number is best performance)
 - Decimal number provides most precision, default with decimals, can have rounding issues
 - Date is internally stored as a decimal number rounded to seconds of a day

15. What are role playing dimensions?

- a. What options are available for handling role playing dimensions in SSAS Tabular and Power BI?

- A fact table can have multiple relationships to a single dimension BUT only one relationship can be active at a time (e.g. Order Date, Ship Date, Due Date), the others are passive
- The USERELATIONSHIP function can be used to make a specific relationship active
 - CALCULATE([MySalesMeasure],USERELATIONSHIP(Sales[DueDate],’Date’[Date]))
- Alternatively, separate, dedicated dimension tables can be used with their own active relationships to the fact table. These can be created via SQL or new Calculated Tables Function

16. What are semi-additive measures and how can they be addressed in Power BI and SSAS Tabular?

- Semi-additive measures are measures which cannot be summed across all dimensions. Examples include ‘snapshot’ fact tables of inventory or bank statements – a metric must reflect a specific point in time
- DAX functions such as LASTNONBLANK(), FIRSTNONBLANK(), and

17. What is a calculated column in Power BI and SSAS Tabular and why would you use or choose not to use them?

- Calculated Columns are DAX expressions that are computed during the model's processing/refresh process for each row of the given column and can be used like any other column in the model
- Calculated columns are not compressed and thus consume more memory and result in reduced query performance. They can also reduce processing/refresh performance if applied on large fact tables and can make a model more difficult to maintain/support given that the calculated column is not present in the source system

18. How is data security implemented in Power BI and SSAS Tabular models?

a. What is dynamic security and how is it applied at a high level?

- Power BI and SSAS Tabular can apply Row Level Security roles to models
 - A DAX expression is applied on a table filtering its rows at query time
- Dynamic security involves the use of USERNAME functions in security role definitions. Typically a table is created in the model that relates users to specific dimensions and a role

19. In loading a data model, why would you retrieve from source tables directly or from views?

- It's recommended to create or use a layer of views rather than source tables directly. The views protect the model from changes in the source tables such as the removal or addition of columns. It creates a layer of indirection and improves manageability

20. What are some of the new modeling features in SSAS Tabular 2016 and why would you use them?

- Bi-Directional Crossfiltering
- Calculated Tables
- Display Folders
- Parallel Partition Processing

21. What are the implications and considerations of the different relationship cross-filtering behaviors in Power BI and SSAS Tabular?

a. For example, if bi-directional relationships are not used, how can I have a metric for the count of distinct product categories (related to a Sales table)?

- Can use DAX and the CALCULATE (or CALCULATETABLE) function to execute a metric in the filter context of a fact table

22. What are many-to-many relationships and how can they be addressed in Power BI and SSAS Tabular?

- Many to Many relationships involve a bridge or junction table reflecting the combinations of two dimensions (e.g. doctors and patients). Either all possible combinations or those combinations that have occurred.
- Bi-Directional Crossfiltering relationships can be used in PBIX and SSAS Tabular 2016
- CROSSFILTER function can be used in Power Pivot for Excel
- DAX can be used per metric to check and optionally modify the filter context

23. What rules must be followed in creating a relationship between two tables in Power BI and SSAS Tabular?

- If the source tables don't meet this criteria, how might you resolve to create the relationship

- One of the two relationship columns must uniquely identify the rows of one of the tables
- Relationships are defined between a single column on each table, not multiple columns
- Data types of each relationship column must be equal
- It may be necessary to create relationship column via concatenating multiple columns
 - Columns could be concatenated directly or modified prior to being merged

24. How do you ensure proper sorting order of a column's values such as calendar month?

- You can import a column of type whole number that matches the cardinality of the attribute (e.g. January = 1) and sort the attribute by this column.
 - This supporting column is generally hidden from client tools

25. What are perspectives in SSAS and provide an example of how they would be used?

- Perspectives provide a limited, defined view of the overall model to only include specific tables, columns, and measures.
- Perspectives can be used to simplify the user experience – users in the finance organization could connect to a 'finance' perspective instead of the entire model and this perspective would only contain finance related tables, columns, and metrics

26. Should the default Summarize By property of columns be used?

- For example, if a sales amount column defaults to Summarize by Sum, is it okay to allow this column to be used in reports and dashboards?

- No, measures should be created which explicitly define the aggregation/calculation to be applied to the model.
- Measures can be referenced/re-used in other metrics, can contain defined formatting, can be used in display folders, and can simplify reporting/analysis experience relative to default summarization

27. Why would you choose to hide certain columns or tables from client tools?

- Surrogate key and metadata tables and columns may not be needed by business users and make the model less intuitive. Only columns needed for self-service data analysis and report development should be exposed to client tools.

28. Why might you have a table in the model without any relationships to other tables?

- A disconnected table might be used to present the user with parameter values to be exposed and selected in slicers (e.g. growth assumption, Top X)
 - DAX metrics could retrieve this selection and use it with other calculations/metrics
- A disconnected table may also be used as a placeholder for metrics in the user interface.
 - It may not contain any rows of data and its columns could be hidden but all metrics

29. What are some differences in data modeling between Power BI Desktop and Power Pivot for Excel?

- Power BI Desktop supports bi-directional crossfiltering relationships, security, calculated tables, and Direct Query options.
- Power Pivot for Excel has single direction (one to many) relationships, calculated columns only, and supports import mode only. Security roles cannot be defined in Power Pivot for Excel.

30. What are some ways that Excel skills, workbooks, and experience can be leveraged with Power BI?

- The Power BI Publisher for Excel
 - Can be used to pin Excel items (charts, ranges, pivot tables) to Power BI Service
 - Can be used to connect to datasets and reports stored in Power BI Service
- Excel workbooks can be uploaded to Power BI and viewed in the browser like Excel Services
- Excel reports in the Power BI service can be shared via Content Packs like other reports
- Excel workbooks (model and tables) can be exported to service for PBI report creation
- Excel workbook Power Pivot models can be imported to Power BI Desktop models

31. How can data models in Power Pivot, Power BI Desktop, and SSAS Tabular be migrated? For example, can a model in one tool be imported to another?

- A Power Pivot for Excel model can be imported to Power BI Desktop including its Power Query components (if any) and any Power View report pages
- A Power Pivot for Excel model can be imported to SSAS Tabular if it does not use Power Query/Query Editor
- A Power BI Desktop model cannot currently be imported/migrated to either SSAS Tabular 2016 or Power Pivot for Excel 2016 given its use of Query Editor/Power Query.
 - This is being addressed in the next release of SSAS Tabular and the 1400 compatibility level

32. What query languages are used by Microsoft Excel and Power BI when sending queries to SSAS Tabular?

a. What are the implications or considerations of this?

- MS Excel sends MDX queries, Power BI Desktop and Power BI Service sends DAX queries
- Greater performance can be obtained via Power BI given enhancements to DAX in 2015-2016 including new table functions and variables and other optimizations

33. What are some considerations in designing the size of the partitions in an SSAS Tabular model?

a. What are benefits and costs of larger and smaller partitions?

- The partitions should be small enough to improve manageability and processing performance to support necessary refresh windows BUT should be large enough to avoid hurting query performance (CPU parallelization and compression)
 - Each partition has its own row segment(s) and each row segment is assigned a CPU thread at query time if accessed
 - Too many partitions can diminish compression and cause unnecessary aggregation of CPU thread results (or run out of CPU threads)

Power Query/Query Editor

1. Why would you choose to use SQL or the Query Editor (or Power Query for Excel) when retrieving data into the model?
 - a. What are some tradeoffs between the different approaches?

- The Query Editor and Power Query can quickly produce ETL packages to merge, transform, filter, and cleanse source data
 - However, PBIX and Excel files which use Query Editor and Power Query, respectively, cannot be migrated to SSAS Tabular without removing these queries
 - Depending on the query folding applied and the client machine, queries can be slow
- SQL requires a knowledge of the ANSI standard and source-specific syntax (no GUI)
 - However, using SQL only in Power Pivot for Excel allows the model to be migrated to SSAS Tabular

2. What is Query Folding in Power Query and the Power BI Query Editor?
 - a. Why is it important?

- Query folding is when steps defined in Power Query/Query Editor are translated into SQL and executed by the source database rather than the client machine
- It's important for processing performance and scalability given limited resources on the client machine

3. What are some common Power Query/Editor Transforms?

- Changing Data Types, Filtering Rows, Choosing/Removing Columns, Grouping, Splitting a column into multiple columns, Adding New Columns

4. Can you describe options available for merging and appending multiple Power Queries together?

- Two power queries can be joined/merged via Inner, Left, Right, Full Outer, and Left anti and Right Anti
 - Tables can be merged/joined based on multiple columns

5. What's the difference between Basic and Advanced Filter Rows in Power Query/Query Editor

- Basic: can only operate on one column and can use at most two conditions separated by an AND or an OR
- Advanced: can apply filter clauses on any column of the table and can apply many filtering conditions

6. Can SQL and Power Query/Query Editor be used together?

a. Why would this be a good practice?

- Yes, a SQL statement can be defined as the source of a Power Query/M function for additional processing/logic. This would be a good practice to ensure that an efficient database query is passed to the source and avoid unnecessary processing and complexity by the client machine and M function

7. Is it possible to Pivot or Unpivot data with Power Query/Query Editor?

a. Can you give an example of unpivoting source data?

- Yes, there's built-in support for pivoting/un-pivoting columns in Power Query/Query Editor
- It may be necessary to unpivot a table with financial periods, months, or dates as columns into a narrower table for loading into the model

8. What are functions and how/why would they be created or used?

- Functions are essentially queries (M functions) that are invoked by other queries and receive parameter values.
 - For example, a function could receive a directory path as its parameter and process multiple files in a directory

9. What are query parameters and Power BI templates?

a. Give an example of how they might be used?

- Query parameters can be used to provide users of a local Power BI Desktop report with a prompt to specify the values they're interested in.
 - The parameter selection can then be used by the query and calculations
- PBIX files can be exported as Templates (PBIT files)
 - Templates contain everything in the PBIX except the data itself
- Parameters and templates can make it possible to share/email smaller template files and limit the amount of data loaded into the local PBIX files, improving processing time and experience

DAX

1. What are some of the most common, basic DAX Functions you use or have used?
 - a. Can you describe some of the more complex implementations or use cases of DAX you've supported?

- SUM, MIN, MAX, AVG, COUNTROWS, DISTINCTCOUNT
- IF, AND, OR, SWITCH
- ISBLANK, ISFILTERED, ISCROSSFILTERED
- VALUES, ALL, FILTER, CALCULATE,

2. Can you define or explain the VALUES() function and the ALL() function?
 - a. What do they return? What operation do they perform over their input?

- VALUES returns a table of a single column of distinct values including a value for blanks if any for the given filter context (at query run time)
- ALL returns a table and removes any existing filters applied to the table

3. How is the FILTER function used?
 - a. What are some considerations with using it or not using it?

- The FILTER function returns a table with a filter condition applied for each of its source table rows. The FILTER function is rarely used in isolation, it's generally used as a parameter to other functions such as CALCULATE
- FILTER is an iterator and thus can negatively impact performance over large source tables
- Complex filtering logic can be applied such as referencing a measure in a filter expression
 - FILTER(MyTable,[SalesMetric] > 500)

4. What is special or unique about the CALCULATE and CALCULATETABLE functions?
 - a. What are some limitations of these functions?

- These are the only functions that allow you modify filter context of measures or tables
 - Add to existing filter context of queries
 - Override filter context from queries
 - Remove existing filter context from queries
- Limitations
 - Filter parameters can only operate on a single column at a time
 - Filter parameters cannot reference a metric

5. Describe DAX as a Query Language structure
 - a. What clauses or terms are required? What optional clauses?
 - b. What are some common functions when using DAX as a query language?

- EVALUATE is the only required term
 - Must return a table
- DEFINE is optional for defining measures at query run time
- ORDER BY and START AT clauses are optional
- Common Query Functions include: SUMMARIZE(), ADDCOLUMNS(), CALCULATETABLE(), ROW(), SUMMARIZECOLUMN(),

6. What is the common table function for grouping data?

a. Describe some examples of GROUP BY like queries via DAX

- SUMMARIZE()
 - Main group by function in SSAS Tabular 2012
 - Recommended practice is to specify table and group by columns but not metrics – use ADDCOLUMNS
- SUMMARIZECOLUMNS
 - New group by function for SSAS Tabular 2016 and Power BI Desktop; more efficient
 - Specify group by columns, table, and expressions

7. What are some new DAX Functions available in SSAS 2016, Power Pivot for Excel 2016, and Power BI?

- UNION, INTERSECT, EXCEPT, NATURALINNERJOIN, NATURALLEFTOUTERJOIN, SUMMARIZECOLUMNS, ISEMPTY,
- VAR (Variables)
- GEOMEAN, MEDIAN, DATEDIFF

8. What are some benefits and use cases of using Variables in DAX

a. What are some limitations if any?

- By declaring and evaluating a variable, the variable can be re-used multiple times in a DAX expression thus avoiding additional queries of the source database
- Variables can make DAX expressions more intuitive/logical to interpret
- Variables are only scoped to their measure or query, they cannot be shared among measures or queries or defined at the model level

9. Are you familiar with the CONTAINS function?

a. Why might it be more efficient than other approaches?

- CONTAINS is similar to an EXISTS function in SQL and returns TRUE if a single row is found in the input table that matches one of the values
- By only checking a large table for the existence of one row that meets the match condition, CONTAINS can be more efficient than a query that returns all the rows from the table that match the condition

10. How would you create trailing X month metrics via DAX against a non-standard calendar?

- Solution will involve:
 - CALCULATE function to control (take over) filter context of measures
 - ALL to remove existing filters on the date dimension
 - FILTER to identify which rows of the date dimension to use
 - Alternatively CONTAINS may be used
- CALCULATE(FILTER(ALL('DATE'),.....))

Power BI Report Authoring

1. What are the differences between a Power BI Dataset, a Report, and a Dashboard?

- Dataset: The source used to create reports and visuals/tiles
 - A data model (local to PBIX or XLSX) or model in an Analysis Services Server
 - Data could be inside of model (imported) or a Direct Query connection to a source
- Report: An individual Power BI Desktop file (PBIX) containing one or more report pages
 - Built for deep, interactive analysis experience of a given dataset (filters, formatting)
 - Each Report is connected to one dataset (which may be
 - Each page containing one or more visuals or tiles, some of which
- Dashboard: a collection of visuals or tiles from different reports and, optionally, a pinned
 - Built to aggregate primary visuals and metrics from multiple datasets

2. What are the three different levels or scopes of filters in Power BI Desktop?

a. Give an example of using each of them?

- Visual Level Filters: a filter applied to a specific visual or tile in a report
- Page level filters: a filter applied to all visuals or tiles in a report
- Report Level filters: a filter applied to every tile/visual in the report (PBIX file)

3. Are there any limitations or restrictions to these filters in Power BI Desktop?

- The value of a metric cannot be used at the Page or Report level scopes
- A metric can only be used for a specific visual

4. What does Edit Interactions do at a high level in Power BI Desktop?

a. Why might a Power BI developer or report author turn on or off Interactions?

- Edit interactions can be used to define which tiles/objects receive and respect filter selections made on other objects and the type of cross-filtering behavior per visual/tile

5. What are some of the capabilities available in the Analytics pane in Power BI Desktop?

- Add Min, Max, Average, Median, Constant, and Percentile lines to support graphs
- Forecasting on line charts

6. What are the three Edit Interactions options of a visual tile in Power BI Desktop? Explain them.

- Filter, Highlight, and None
 - Filter: completely filter a visual/tile based on the filter selection of another visual/tile
 - Highlight: highlight only the related elements on the visual/tile, only gray out the non-related items
 - None: Ignore the filter selection from another tile/visual

7. How can a Power BI report author better prepare a model for Cortana Q&A?

- The author can apply 'Synonyms' to the model in the modeling window to associate common business terms with tables and columns
- Synonyms can be applied to tables, columns, and metrics

8. What does drill mode do in Power BI Desktop/Service?

- Drill mode allows a user to view all the child items of a parent in a visual by selecting on the parent
 - For example, with drill mode enabled on a column chart showing product categories, clicking on the 'Bike Category' will show the metric broken out by the subcategories of bike category if this additional attribute is included in the visual

9. What's the difference between going to the next level in a hierarchy and Expand All down one level in a hierarchy? (the two drilling options on the left hand side of visuals in Power BI)

a. How is this different than drill mode?

- Going to the next level displays all the child items, not just those associated with a single parent item like with drill mode
- 'Expand All down one level' concatenates the parent item label with the child item label such that, when viewing the subcategories you can still see the parent category it's associated with

10. What are some of the differences in report authoring capabilities between using a live or direct query connection such as to an Analysis Services model relative to working with a data model local to the PBI Desktop file?

- With a data model local to the PBIX file (or Power Pivot workbook), the author has full control over the queries, the modeling/relationships, the metadata and the metrics
- With a live connection to an Analysis Services database (cube) the user cannot create new metrics, import new data, change the formatting of the metrics, etc – the user can only use the visualization, analytics, and formatting available on the report canvas
- With a direct query model in PBIX, to SQL Server relational for example, the author has access to the same features (and limitations) available to SSAS Tabular 2016 Direct Query mode
 - Only one data source (one database on one server) may be used, certain DAX functions are not optimized, and the user cannot use Query Editor functions that cannot be translated into SQL statements

11. What are some of the new features and recent enhancements to report authoring in Power BI Desktop (late 2016, early 2017)? Name 3-4 and a brief description.

- Date Slicer, Report Gridlines and Snap to Grid, Data Label Improvements, Date Axis Range Formatting, Dropdown Slicer, Mobile Reports View
- Grouping, Binning, Top N Filter, Include/Exclude Data Points, Aggregations for Dates and Text
- Table and Matrix Conditional Formatting, Column Formatting, Clustering, Word Wrap

Power BI Service and Collaboration

1. What are Power BI workspaces used for and what are their main configuration settings?

- Workspaces are used by teams to collaborate on relevant reports and dashboards and as an additional security layer to restrict access to datasets, reports, and dashboards
- Workspaces can be public or private and members can either view only or edit their content
 - If view only, only group admins can edit content

2. What are some features provided to each group?

- Each group workspace is allocated 10GB of OneDrive for Business storage that can be used if the user O365 subscriptions include OneDrive for Business

3. What are some scenarios for using group workspaces?

- User acceptance testing – read only group workspaces
- Business Team Self-Service – (example: finance team has their own workspace for editing content)

4. What are the two types of Power BI content packs and how are they used?

a. Please provide an example

- Organizational Content Packs: a collection of dashboards, reports, and datasets made available for an organization's users to consume/access in Power BI
 - Control who can access; recipients have option to edit local copies
 - Changes can be re-published to recipients
- Service Content Packs: pre-built Power BI reports, dashboards, and datasets by third parties for connecting to and analyzing 3rd party services

5. How can OneDrive for Business be used with Power BI Service?

a. What are advantages and provide a short example?

- With an Office 365 subscription inclusive of OneDrive, a user can save PBIX or XLSX file to OneDrive
- Multiple versions of a PBIX file could be stored in OneDrive for Business
- PBIX files could be imported to Power BI Service from OneDrive for Business for other users to access
- An Excel file could remain stored in OneDrive for Business and the author could share access with other users. The file would appear as an Excel report in Power BI and Power BI could provide the scheduled refresh

6. What is the on-premises data management gateway?

a. What are the two kinds of gateways and what are there differences and limitations?

- Gateway acts a bridge between on-premises data sources and Azure cloud services
- Personal Gateway:
 - Import Only, Power BI Service Only, No central monitoring/managing
 - Can only be used by one person (personal); can't allow others to use this gateway
- On-Premises Gateway:
 - Import and Direct Query supported
 - Multiple users of the gateway for developing content
 - Central monitoring and control

7. How do I configure a data alert in Power BI service?

- Build a PBIX report with one of the following visuals: tile with number, gauge, KPI
- Pin the visual to a Power BI Dashboard in the service
- Configure an alert on the tile

8. Are there options available for embedding Power BI into other applications or portals?

- Yes, the Power BI REST APIs can be used to embed tiles from either reports and dashboards into apps and portals.
 - The iFrame URLs are available from the APIs
 - Authentication of users is handled through Power BI (same security)
- Power BI embedded can also be used

9. What are the mobile capabilities available with Power BI?

- The Power BI mobile application is available for all mobile platforms including Windows 10
- Power BI reports can be optimized for mobile layouts such that when accessed via mobile device
- The Power BI mobile application can also be used to access on-premises SSRS paginated and mobile reports

10. How does SSRS integrate with Power BI?

- Certain SSRS Report items such as charts can be pinned to Power BI dashboards
 - Clicking the tile in Power BI dashboards will bring the user to the SSRS report
 - A subscription is created to keep the dashboard tile refreshed
- Power BI reports will soon be able to be published to SSRS portal
 - Not available in SSRS 2016, available in upcoming next release of SQL Server

11. What is the Power BI Publisher for Excel?

a. Provide some examples of how it could be used?

- You can use Power BI publisher for Excel to pin ranges, pivot tables and charts to Power BI
 - The user can manage the tiles – refresh them, remove them, in Excel
 - Pinned items must be removed from the dashboard in the service (removing in Excel only deletes the connection)
- The Power BI Publisher for Excel can also be used to connect from Excel to datasets that are hosted in the Power BI Service
 - An Excel pivot table is generated with a connection (ODC file) to the data in Azure
- The Publisher installs all necessary drivers on local machine to establish connectivity