EZConvertBI Tableau Analyzer Report - User Guide

Section 1: Executive Summary & Quick Navigation

Report Purpose & Context

This Excel report provides a comprehensive analysis of your Tableau Server environment specific to a site, designed to support:

- **Migration Planning**: Assess conversion feasibility and effort estimation
- **Inventory Analysis**: Complete catalog of all Tableau objects and dependencies
- **Optimization Opportunities**: Identify consolidation and rationalization possibilities

The report is generated from a live Tableau Server extraction and contains 18 interconnected sheets with detailed metadata about your environment.

Key Metrics Dashboard (Summary Sheet)

The Summary sheet provides essential context:

Metric	What It Tells You	Business Impact
Extraction Date	When data was captured	Ensures analysis relevance
Tableau Server URL	Source environment	Confirms correct server analyzed
Complexity Categorization	Dashboard difficulty distribution	Migration effort estimation
Adjusted Complexity	Refined complexity after analysis	More accurate planning
Total Objects	Environment size (Workbooks/Dashboards/Views)	Resource planning scope

Complexity Categories Explained:

- **Low**: Simple dashboards, minimal custom logic
- **Medium**: Higher number of visualizations, some calculated fields
- **High**: Complex/high number of visualizations, multiple data sources, lots of calculations and parameters
- **Very High**: High complexity plus advanced features like blended relationships, extensive custom calculations

Navigation by Role

Executive Path (10-minute overview)

1. **Summary Sheet** → Overall environment health
2. **Conversion Summary** → Migration feasibility
3. **Rationalization Summary** → Optimization opportunities

Technical Path (detailed analysis)

1. **Summary Sheet** → Context
2. **Workbooks/Dashboards/Sheets** → Core inventory
3. **DB Connections** → Data dependencies
4. **Calculated Fields** → Custom logic complexity

Migration Planning Path

1. **Summary** → Scope understanding
2. **Conversion Summary** → What can/cannot be converted
3. **Charts Report** → Visualization-level conversion details
4. **Rationalized Dashboards** → Consolidation opportunities

Sheet Categories Overview

Core Inventory (Start Here)

- **Summary**: High-level metrics and complexity overview
- **Workbooks**: Master catalog of all workbook files
- **Dashboards**: Individual dashboard details with complexity scores
- **Views**: All visualizations and their properties
- **Sheets**: Worksheet-level breakdown

Data Architecture

- **Published Data Sources**: Centralized, reusable data connections
- **Embedded Data Sources**: Workbook-specific data sources
- **DB Connections**: Database and system connectivity details
- **Parameters**: Dynamic filters and user controls

Technical Details

- **Calculated Fields**: Custom business logic and formulas
- **Chart Types**: Visualization type distribution analysis
- **Table Relations**: Data model relationships and joins
- **Blended Relationships**: Cross-source data blending logic
- **Dataset Definitions**: Underlying data structure details

Migration Planning

- **Conversion Summary**: Overall migration feasibility by workbook
- **Charts Report**: Visualization-level conversion readiness
- **Rationalization Summary**: Dashboard optimization analysis
- **Rationalized Dashboards**: Specific consolidation recommendations

Understanding Sheet Relationships

Key ID Fields for Cross-Referencing:

- `workbook_id` → Links Workbooks to Dashboards/Views/Datasources etc.
- `dashboard_id` → Links Dashboards to Charts/Views
- `view_id` → Links Views to specific visualizations
- `datasource_id` → Links data sources across sheets/dataset definitions/calculated fields

Common Navigation Patterns:

1. **Workbook → Dashboard → Views**: Drill down from high level to details
2. **Data Source/Workbook → Calculated Fields**: Find custom logic for specific Datasource/Workbook
3. **DB Connections → Data Sources**: Understand data pipeline
4. **Conversion Summary → Charts Report**: Migration planning workflow

Quick Start Checklist

For First-Time Users:

- Check **Summary sheet** extraction date
- Review **complexity distribution** in Summary
- Scan **Workbooks sheet** for environment scope
- Check **Conversion Summary** for migration feasibility
- Identify high-priority items in **Dashboards sheet**

Section 2: Core Inventory Analysis

Understanding Your Tableau Environment

This section covers the foundational sheets that catalog all objects in your Tableau Server. Use these sheets to understand the scope and structure of your environment.

2.1 Workbooks Sheet

Purpose: Master inventory of all workbook files in your Tableau Server

Key Columns:

- `workbook_id`: Unique identifier (links to all other sheets)
- `workbook_name`: Display name of the workbook
- `project_name`: Tableau project location
- `owner_name`: Primary owner/creator
- `createdAt/updatedAt`: Timestamps for lifecycle tracking
- `sheetCount/dashboardCount`: Content volume indicators

How to Use:

1. **Scope Assessment**: Count total workbooks to understand environment size
2. **Owner Analysis**: Identify key stakeholders by owner_name
3. **Project Mapping**: Group by project_name for organizational structure
4. **Age Analysis**: Use createdAt/updatedAt to identify stale content
5. **Content Volume**: Use sheetCount/dashboardCount for complexity indicators

Cross-References:

- Links to **Dashboards** via `workbook_id`
- Links to **Views** via `workbook_id`
- Links to **Published/Embedded Datasources** via `workbook_id`
- Links to Parameters/Calculated Fields/and other sheets via `workbook_id`
- Owner information connects to organizational planning

Common Patterns:

- High sheetCount + low dashboardCount = worksheet-heavy workbooks
- Recent updatedAt = actively maintained content

2.2 Dashboards Sheet

Purpose: Detailed catalog of all dashboards with complexity analysis

Key Columns:

- `dashboard_id`: Unique identifier (primary key)
- `dashboard_luid`: Tableau's internal ID
- `dashboard_name`: Display name
- `complexity/complexity_score`: Difficulty assessment (see Section 1)
- `adj_complexity/adj_complexity_score`: Refined complexity after analysis
- `sheetCount`: Number of worksheets in dashboard
- `workbook_id`: Parent workbook reference
- `calc_field_count`: Number of calculated fields used

How to Use:

1. **Migration Prioritization**: Sort by complexity to identify easy wins vs challenges
2. **Effort Estimation**: Use complexity_score for resource planning
3. **Content Analysis**: High sheetCount = complex multi-view dashboards
4. **Dependency Planning**: calc_field_count indicates custom logic complexity

Understanding Complexity Scores:

- **Low (0-100)**: Straightforward migration
- **Medium (101-200)**: Moderate effort required
- **High (201-300)**: Significant migration work
- **Very High (300+)**: Complex migration requiring manual intervention

Cross-References:

- Links back to **Workbooks** via `workbook_id`
- Links to **Views** and **Sheets** via `dashboard_id`
- Links to **Chart Types** for visualization breakdown

Red Flags to Watch:

- Very High complexity with high calc_field_count
- Large sheetCount with recent updatedAt (heavily used complex content)

2.3 Views Sheet

Purpose: Comprehensive inventory of all individual visualizations

Key Columns:

- `view_id`: Unique identifier
- `view_luid`: Tableau's internal ID
- `view_name`: Display name of the view
- `view_type`: Type of visualization (Chart, Dashboard, etc.)
- `view_path`: Location path in Tableau Server
- `workbook_id`: Parent workbook reference
- `createdAt/updatedAt`: Lifecycle timestamps

How to Use:

1. **Visualization Inventory**: Complete catalog of all visual content
2. **Usage Patterns**: Identify most common view_types
3. **Navigation Mapping**: Use view_path for server structure understanding
4. **Content Freshness**: Track updatedAt for maintenance insights

View Types Explained:

- **Sheet**: Individual worksheet/visualizations
- **Dashboard**: Multi-chart dashboards/ Tableau story presentations

Cross-References:

- Links to **Workbooks** via `workbook_id`
- Links to **Chart Types** for detailed visualization analysis
- Path information connects to server organization

2.4 Sheets Sheet

Purpose: Worksheet-level details for technical analysis

Key Columns:

- `sheet_id`: Unique identifier
- `sheet_luid`: Tableau's internal ID
- `sheet_name`: Worksheet name
- `path`: Full path in workbook structure
- `createdAt/updatedAt`: Timestamps
- Additional technical metadata columns

How to Use:

1. **Technical Inventory**: Worksheet-level granular analysis
2. **Structure Mapping**: Understand workbook internal organization
3. **Content Tracking**: Individual worksheet lifecycle management
4. **Migration Planning**: Worksheet-specific conversion requirements

Cross-References:

- Connects to parent **Dashboards** and **Workbooks**
- Links to **Blended Relationships** used in specific sheets
- Technical foundation for detailed migration planning

Section 2 Summary

Key Relationships: Workbooks → Dashboards → Views/Sheets → Technical Details

Migration Planning Workflow:

1. Start with **Workbooks** for overall scope
2. Analyze **Dashboards** for complexity assessment
3. Review **Views** for visualization inventory
4. Use **Sheets** for technical implementation details

Section 3: Data Architecture Analysis

Understanding Your Data Layer

This section covers the data foundation of your Tableau environment. Understanding data sources, connections, and parameters is critical for migration planning and dependency mapping.

3.1 Published Data Sources Sheet

Purpose: Catalog of centralized, reusable data connections shared across workbooks

Key Columns:

- `published_data_source_id`: Unique identifier
- `published_data_source_luid`: Tableau's internal ID
- `published_data_source_name`: Display name
- `hasExtracts`: Whether data is extracted locally
- `hasUserReference`: If user authentication is required
- `createdAt/updatedAt`: Lifecycle timestamps
- `project_name`: Location in Tableau Server
- `owner_name`: Data source owner/maintainer

How to Use:

1. **Centralization Assessment**: High count indicates good data governance
2. **Technology Inventory**: Group by project for migration planning
3. **Extract Analysis**: hasExtracts = true indicates offline data copies
4. **Dependency Mapping**: Shared data sources affect multiple workbooks
5. **Security Planning**: hasUserReference shows authentication requirements

Migration Implications:

- **Extract-based sources**: May need data refresh strategy in target platform
- **Live connections**: Require network connectivity planning
- **Shared sources**: Changes affect multiple workbooks simultaneously
- **User authentication**: Security integration requirements

Cross-References:

- Used by multiple **Workbooks** and **Embedded Datasources**
- Links to **Embedded Datasources** for underlying connectivity
- Owner/Project connects to organizational structure

Empty Sheet Interpretation: If this sheet has headers but no data rows, your environment uses primarily embedded data sources rather than centralized published ones. This indicates potential governance opportunities.

3.2 Embedded Data Sources Sheet

Purpose: Workbook-specific data connections that are not shared

Key Columns:

- `embedded_data_source_id`: Unique identifier
- `embedded_data_source_name`: Internal name
- `embedded_data_source_type`: Technology type
- `hasExtracts`: Local data extraction flag
- `workbook_id`: Parent workbook reference
- `columns`: metadata of data columns
- `query`: custom sql run on the datasource
- `upstreamdatabaseid`: the data connection used (DB Connection)
- `upstreamdatasourceid`: the published datasource being referred
- Technical metadata columns

How to Use:

1. **Decentralization Analysis**: High count may indicate data governance gaps
2. **Workbook Dependencies**: Each embedded source ties to specific workbook
3. **Data Complexity**: Use columns/query/filters/upstreamdatasource for sizing estimates
4. **Technology Spread**: Group by workbook_id/upstreamdatasource for consolidation opportunities

Migration Considerations:

- **Embedded sources**: Easier to migrate (self-contained)
- **Multiple similar sources**: Consolidation opportunities
- **Extract vs Live**: Different migration strategies required
- **Complex data models**: Higher migration effort

Cross-References:

- Links to **Workbooks** via `workbook_id`
- Connects to **DB Connections** for underlying systems
- References **Dataset Definitions** for data structure details

3.3 DB Connections Sheet

Purpose: Underlying database and system connectivity details

Key Columns:

- `database_id`: Unique identifier
- `host`: Server/system hostname
- `connectionType`: Database technology (SQL Server, Oracle, File, etc.)
- `database_name`: Specific database name
- `port`: Network port (if applicable)
- Additional connection parameters

How to Use:

1. **Infrastructure Mapping**: Catalog all backend systems
2. **Technology Assessment**: Group by connectionType for migration planning
3. **Network Planning**: Use host/port for connectivity requirements
4. **Security Review**: Credentials extraction for authentication planning
5. **Consolidation Opportunities**: Identify duplicate connections

Connection Types Explained:

- **Database**: Traditional databases
- **File**: Excel, CSV, or other file-based sources
- **CloudFile**: googlesheets, sharepoint files etc
- **Other**: Rdata, sharepoint lists, jsons etc.

Migration Planning:

- **Database connections**: May require new connection strings
- **File-based**: Need file migration strategy
- **Cloud services**: Authentication and networking considerations
- **Custom connectors**: May need rebuilding in target platform

Cross-References:

- Used by **Published Data Sources** and **Embedded Data Sources**
- Connects to infrastructure and security planning
- Links to **Dataset Definitions** for data model understanding

3.4 Parameters Sheet

Purpose: Dynamic filters, controls, and user input mechanisms

Key Columns:

- `parameter_id`: Unique identifier
- `parameter_name`: Display name
- `alias`: User-friendly label

- `value`: Default or current value
- `multi_values`: Whether multiple selections allowed
- `dataType`: Parameter data type (String, Integer, Date, etc.)
- `workbook_id`: Parent workbook reference
- Additional parameter configuration

How to Use:

1. **Interactivity Assessment**: Count indicates dashboard user engagement level
2. **Migration Complexity**: Parameters often need manual recreation
3. **User Experience**: Parameters with multi_values are more complex
4. **Data Type Planning**: Different types have different migration requirements

Parameter Types:

- **Single Value**: Simple dropdown or input
- **Multi-Value**: Complex selection controls
- **Date Parameters**: Time-based filtering
- **Numeric Parameters**: Range controls and calculations

Migration Considerations:

- **Complex parameters**: May need redesign in target platform
- **Multi-value parameters**: Often require custom development
- **Parameter dependencies**: May affect calculated fields and filters
- **User training**: Parameter behavior may change post-migration

Cross-References:

- Links to **Workbooks** via `workbook_id`
- Used in **Calculated Fields** for dynamic logic
- Affects **Dashboard** interactivity and complexity

Section 3 Summary

Data Architecture Patterns:

- **High Published + Low Embedded**: Good centralized governance
- **High Embedded + Low Published**: Decentralized, potential consolidation opportunities
- **Many DB Connections**: Complex infrastructure dependencies
- **Many Parameters**: High interactivity, complex migration

Migration Planning Workflow:

1. **Published Data Sources**: Plan shared data migration first
2. **DB Connections**: Assess infrastructure requirements

3. **Embedded Data Sources**: Plan workbook-specific data handling
4. **Parameters**: Plan user interaction recreation

Key Dependencies: Published Data Sources → Multiple Workbooks (high impact changes) Embedded Data Sources → Single Workbooks (isolated impact) DB Connections → All Data Sources (infrastructure foundation) Parameters → Dashboard Interactivity (user experience impact)

Section 4: Technical Complexity Analysis

Understanding Migration Effort and Custom Logic

This section covers the technical components that typically require the most effort during migration. These sheets help estimate conversion complexity and identify potential blockers.

4.1 Calculated Fields Sheet

Purpose: Comprehensive inventory of all custom business logic and formulas

Key Columns:

- `calculated_fields_id`: Unique identifier
- `calculated_fields_name`: Field name as it appears to users
- `formula`: The actual calculation logic
- `dataCategory`: Type of data (Categorical, Quantitative, etc.)
- `dataType`: Data format (String, Integer, Real, Date, etc.)
- `workbook_id`: Parent workbook reference
- `datasource_id`: Associated embedded datasource (if applicable)
- `hierarchy`: Level of nesting
- Additional technical metadata

How to Use:

1. **Complexity Assessment**: Count total calculated fields for effort estimation
2. **Formula Analysis**: Review `formula` column for migration complexity
3. **Data Type Planning**: Different dataTypes require different handling
4. **Dependency Mapping**: Use workbook_id/datasource_id for impact analysis, used_in_dash to understand usage.
5. **Business Logic Inventory**: Document custom business rules

Formula Complexity Indicators:

- **Simple**: Basic arithmetic (SUM, AVG, simple IF statements)
- **Medium**: Multiple conditions, date functions, string manipulation
- **Complex**: Nested logic, table calculations, LOD expressions, higher heirachy level

- **Very Complex**: Advanced analytics, custom aggregations, complex nesting and parameters used

Migration Risk Assessment:

- **High formula count**: Significant manual conversion effort
- **Complex nested formulas**: May need redesign in target platform
- **Tableau-specific functions**: Require platform-equivalent mapping
- **Table calculations**: Often need complete rebuilding

Cross-References:

- Links to **Workbooks** and **Datasources** via respective IDs
- Used in **Charts** and **Views** for visualization logic
- May reference **Parameters** for dynamic calculations

Common Patterns:

- High calc_field_count in Dashboards sheet correlates with rows here
- Similar formulas across workbooks indicate consolidation opportunities
- Date/time calculations often need platform-specific adjustments

4.2 Table Relations Sheet

Purpose: Data model relationships and join logic between tables

Key Columns:

- `datasource_id`: Embedded Data source identifier
- `datasource_name`: Embedded Data source display name
- `primary_table`: Main table in relationship
- `secondary_table`: Related/joined table
- `on`: Join condition/key fields
- Additional relationship metadata

How to Use:

1. **Data Model Mapping**: Understand how tables connect in a given datasource
2. **Join Complexity**: Review `on` conditions for migration effort
3. **Dependency Analysis**: Complex relationships affect multiple visualizations
4. **Performance Planning**: Multiple joins may impact target platform performance

Relationship Types:

- **Simple**: Single key joins (easy to migrate)
- **Composite**: Multiple field joins (moderate complexity)

- **Complex**: Calculated or conditional joins (high effort)
- **Cross-database**: Joins across different data sources (architecture impact)

Migration Considerations:

- **Simple relationships**: Usually direct conversion
- **Complex join logic**: May need data model redesign
- **Performance implications**: Target platform may handle joins differently
- **Data preparation**: Some joins may move to ETL layer

Cross-References:

- Links to **Published Data Sources**/**Embedded Data Sources**
- Connects to **DB Connections** for underlying data systems
- Affects **Dashboard** and **View** performance

4.3 Blended Relationships Sheet

Purpose: Cross-data source blending and advanced data combination logic

Key Columns:

- `relationship_id`: Unique identifier for the relationship
- `primary_datasource_id/primary_datasource`: Primary data source details
- `secondary_datasource_id/secondary_datasource`: Secondary data source details
- `joining_columns`: Fields used for blending relationship
- Additional blending configuration

How to Use:

1. **Advanced Logic Inventory**: Blending represents complex data combination
2. **Migration Complexity**: Blended relationships often need complete redesign
3. **Data Architecture Impact**: May require ETL or data warehouse changes
4. **Performance Planning**: Blending can be resource-intensive

Migration Strategies:

- **Data warehouse approach**: Move blending logic to ETL
- **Application layer**: Use **Wavicle's EzConvertBI Converter** to handle blends automatically
- **Data preparation**: Pre-combine data sources

Cross-References:

- Links to multiple **Data Sources** (primary and secondary)
- Affects **Dashboard** complexity significantly

- May reference **Calculated Fields** for blend logic

4.4 Dataset Definitions Sheet

Purpose: Underlying data structure and table schema information

Key Columns:

- `upstreamTable_id`: Unique table identifier
- `upstreamTable_luid`: Tableau's internal ID
- `upstreamTable_name`: Table name
- `schema`: Database schema name
- `database_id`: Database connection id
- `datasource_id`: Embedded datasource id
- `columns`: Table columns
- `query`: Custom query used
- Additional metadata about table structure

How to Use:

1. **Data Structure Mapping**: Complete inventory of all tables used
2. **Schema Analysis**: Understand database organization
3. **Migration Planning**: Table-level conversion requirements

Cross-References:

- Connects to **Data Sources** (embedded)
- Links to **DB Connections** for database context
- Foundation for **Table Relations** analysis

Section 4 Summary

Technical Complexity Indicators:

- **High calculated field count**: Significant custom logic migration effort
- **Many complex chart types**: Visualization redesign needed
- **Complex table relations**: Data model migration challenges
- **Blended relationships**: Architecture redesign likely required

Migration Effort Estimation:

1. **Low Effort**: Basic charts, simple calculations, standard joins
2. **Medium Effort**: Some custom logic, moderate chart complexity
3. **High Effort**: Extensive calculated fields, complex visualizations, blended data
4. **Very High Effort**: Advanced analytics, parameters, custom chart types, complex blending

Risk Assessment Workflow:

1. **Calculated Fields**: Count and complexity analysis
2. **Chart Types**: Conversion possibility review
3. **Table Relations**: Data model complexity assessment
4. **Blended Relationships**: Advanced logic identification

Section 5: Migration Planning

Actionable Conversion Strategy and Optimization

This section covers the analysis outputs that directly support migration decision-making, effort estimation, and optimization opportunities. These sheets synthesize all previous analysis into actionable recommendations.

5.1 Conversion Summary Sheet

Purpose: High-level migration feasibility assessment by workbook

Key Columns:

- `workbook_name`: Workbook identifier
- `dashboard_name`: Dashboard within workbook
- `total_charts`: Number of visualizations
- `converter_supported`: Charts that can be automatically converted
- `service_not_supported`: Charts requiring manual intervention
- `source_missing_mandatory_elements`: Charts missing information from Tableau required for conversion, manual intervention required
- `possible_conversion_rate_for_supported_charts`: Overall convertibility score of supported charts
- Additional conversion metrics

How to Use:

1. **Migration Roadmap**: Sort by conversion_percentage for prioritization
2. **Effort Estimation**: Use manual_conversion_required for resource planning
3. **Quick Wins**: High conversion_percentage = easier migrations
4. **Problem Identification**: Low conversion_percentage = complex projects

Conversion Categories:

- **Converter Supported (80-100%)**: Direct automated conversion of supported automated elements possible using **Wavicle's EzConvertBI Converter**
- **Mostly Supported (60-79%)**: Minor manual adjustments needed
- **Partially Supported (40-59%)**: Significant manual work required

- **Limited Support (0-39%)**: Major redesign or custom development needed

Migration Priority Framework:

1. **High Value + High Conversion**: Priority 1 (quick wins)
2. **High Value + Low Conversion**: Priority 2 (complex but important)
3. **Low Value + High Conversion**: Priority 3 (easy backlog items)
4. **Low Value + Low Conversion**: Priority 4 (consider retirement)

Cross-References:

- Links to **Workbooks** and **Dashboards** for detailed context
- Details expanded in **Charts Report** sheet
- Complexity scores from **Dashboards** sheet provide additional context

Red Flags:

- Zero converter_supported charts/service_not_supported = complete manual migration
- High total_charts + low conversion_percentage = major project
- Many service_not_supported = platform limitation issues

5.2 Chart Report Sheet

Purpose: Detailed breakdown of visualization types and conversion feasibility

Key Columns:

- `workbook_name`: Parent workbook reference
- `dashboard_name`: Dashboard display name
- `sheet_name`: Individual chart/visualization name
- `chart_type`: Tableau chart type (bar, line, map, etc.)
- `conversion_possibility`: Migration feasibility assessment
- Additional visualization metadata

How to Use:

1. **Visualization Inventory**: Complete catalog of chart types used
2. **Conversion Planning**: Use conversion_possibility for migration roadmap
3. **Complexity Mapping**: Different unsupported chart types might have different migration efforts
4. **Standardization Opportunities**: Identify commonly used chart patterns

Chart Type Categories:

- **Basic**: Bar, line, table, pie charts (usually easy conversion)
- **Advanced**: Heat maps, treemaps, scatter plots (moderate effort)

- **Geospatial**: Maps with custom territories (high complexity due to dependancies)
- **Specialized**: Tableau-specific visualizations (may need redesign)

Conversion Possibility:

- **Comments** are provided for each unsupported chart to understand underlying issue
- **Supported charts**: Direct conversion possible
- **Unsupported**: Need alternative visualization approach

Cross-References:

- Links to **Dashboards/Workbooks** via `dashboard_name/workbook_name`
- Connects to **Charts Types** for alternative chart types suggestion
- References underlying **Calculated Fields** and **Data Sources**

5.3 Rationalization Summary and Usage Analysis Sheet

Purpose: Dashboard optimization and consolidation opportunities

Key Columns:

- `dashboard_id`: Unique dashboard identifier
- `dashboard_name`: Dashboard display name
- `complexity/complexity_score`: Current complexity assessment
- `dup_dashboard_id:` Unique dashboard identifier of the duplicated dashboard
- `dup_dashboard_name:` Dashboard display name of the duplicated dashboard
- `matched_sheet_percent:` Matching sheet percent between the dashboards (based on deep analysis of measures, dimensions, calculated fields, datasources, filters etc)
- Other metadata

How to Use:

1. **Optimization Planning**: Identify dashboards that can be eliminated
2. **Consolidation Strategy**: Find merge opportunities to reduce total count
3. **ROI Analysis**: Focus on high-usage dashboards for maximum impact
4. **Governance Improvement**: Eliminate redundant or obsolete content

Rationalization Categories:

- **High Potential**: Significant optimization possible, clear benefits
- **Medium Potential**: Some optimization possible, moderate benefits
- **Low Potential**: Minor optimizations only
- **Optimized**: Already well-designed, minimal changes needed

Consolidation Strategies:

- **Similar Content**: Dashboards with overlapping functionality
- **Same Data Sources**: Dashboards using identical data
- **Related Metrics**: Dashboards showing related business KPIs
- **User Groups**: Dashboards serving same audience

Business Benefits:

- **Reduced Complexity**: Easier migration and maintenance
- **Lower Costs**: Fewer objects to convert and support
- **Better User Experience**: Consolidated, focused dashboards
- **Improved Governance**: Cleaner, more manageable environment

Cross-References:

- Links to **Dashboards** sheet for detailed dashboard information
- Connects to **Rationalized Dashboards** for specific consolidation plans
- References **Data Sources** for consolidation feasibility

Section 5 Summary

Migration Planning Workflow:

1. **Conversion Summary**: Establish overall feasibility and priorities
2. **Charts Report**: Plan detailed technical conversion approach
3. **Rationalization Summary**: Identify optimization opportunities

Decision Framework:

- **Convert As-Is**: High conversion percentage, low rationalization opportunity
- **Optimize Then Convert**: Medium conversion, high rationalization potential
- **Redesign**: Low conversion percentage, consider fundamental changes during conversion
- **Retire**: Low conversion, low usage, high maintenance

Key Metrics for Executive Reporting:

- **Total Conversion Percentage**: Overall migration feasibility
- **Manual Effort Required**: Resource planning metric
- **Rationalization Opportunities**: Cost reduction potential
- **Consolidation Count**: Environment simplification measure

Risk Mitigation:

- **Low Conversion Dashboards**: Plan additional time and resources
- **Not Supported Charts**: Develop alternative visualization strategies
- **High Complexity Consolidations**: Phase implementation carefully
- **User Impact**: Comprehensive change management planning

Final Planning Questions:

1. Which dashboards are business-critical and cannot be retired?
2. What is the timeline for user training on consolidated dashboards?
3. How will you handle dashboards with "Not Supported" chart types?

Section 6: Common Workflows

Step-by-Step Guidance for Different Analysis Scenarios

This section provides practical workflows for common analysis scenarios. Follow these step-by-step guides based on your role and objectives.

Executive Review Workflow

Time Required: 15-20 minutes
Audience: Executive leadership, project sponsors
Goal: High-level understanding of migration scope and feasibility

Step 1: Environment Overview

- Open **Summary Sheet**
- Verify extraction date and Tableau Server URL
- Record total counts: Workbooks, Dashboards, Views
- Review complexity distribution (Low/Medium/High/Very High percentages)

Step 2: Migration Feasibility Assessment

- Open **Conversion Summary Sheet**
- Sort by `conversion_percentage` (descending)
- Calculate overall statistics:
    - Average conversion percentage across all workbooks
    - Count of workbooks with >80% conversion (easy wins)
    - Count of workbooks with <40% conversion (major challenges)

Step 3: Cost-Benefit Analysis

- Open **Rationalization Summary Sheet**
- Count dashboards with "High Potential" rationalization opportunities
- Open **Rationalized Dashboards Sheet**
- Count total consolidation recommendations
- Calculate potential dashboard reduction percentage

Step 4: Risk Assessment

- Return to **Conversion Summary Sheet**
- Identify workbooks with high `manual_conversion_required` counts
- Note any workbooks with zero `converter_supported` charts
- Cross-reference with **Dashboards Sheet** complexity scores

Executive Summary Template:

- Environment Size: [X] workbooks, [Y] dashboards, [Z] views
- Overall Conversion Feasibility: [X]% average across workbooks
- Quick Wins: [X] workbooks with >80% conversion rate
- Major Challenges: [X] workbooks requiring significant manual effort
- Optimization Opportunity: [X]% dashboard reduction through consolidation
- Primary Risk Factors: [list top 3 challenges]

Migration Planning Workflow

Time Required: 2-3 hours
Audience: Migration project managers, technical leads
Goal: Detailed migration strategy and resource planning

Step 1: Scope Definition

- **Summary Sheet**: Record environment metrics
- **Workbooks Sheet**: Create project groupings by `project_name` or `owner_name`
- **Dashboards Sheet**: Export dashboard list with complexity scores
- Establish migration phases based on complexity distribution

Step 2: Technical Feasibility Analysis

- **Conversion Summary Sheet**:
  - Sort by `conversion_percentage`
  - Create three buckets: High (>70%), Medium (40-70%), Low (<40%)
  - Flag workbooks with high `manual_conversion_required` counts
- **Charts Report Sheet**:
  - Filter by `conversion_possibility` = "Not Supported"
  - Document alternative approaches for unsupported chart types
  - Group similar conversion challenges for batch handling

Step 3: Resource Estimation

- **Calculated Fields Sheet**: Count by workbook for complexity assessment
- **Charts Report Sheet**: Sum migration effort by workbook
- **Blended Relationships Sheet**: Identify workbooks requiring data architecture changes
- Create effort matrix: [Workbook] x [Estimated Hours] x [Skill Level Required]

Step 4: Dependency Mapping

- **Published Data Sources Sheet**: List shared data dependencies
- **DB Connections Sheet**: Map infrastructure requirements
- **Table Relations Sheet**: Identify complex data models
- Create dependency diagram showing critical path items

Step 5: Optimization Planning

- **Rationalization Summary Sheet**: Prioritize optimization opportunities
- **Rationalized Dashboards Sheet**: Plan consolidation sequence
- Balance optimization benefits vs. migration timeline impact

Step 6: Risk Mitigation Planning

- Low conversion workbooks: Plan additional resources
- Blended relationships: Consider ETL alternatives
- Unsupported chart types: Develop alternative designs
- Complex calculated fields: Plan business logic validation

Migration Plan Template:

- Phase 1: [High conversion workbooks - X weeks]
- Phase 2: [Medium conversion workbooks - Y weeks]
- Phase 3: [Low conversion workbooks - Z weeks]
- Resource Requirements: [X developers, Y analysts, Z weeks each]
- Critical Dependencies: [list infrastructure/data requirements]
- Risk Mitigation: [specific plans for top 5 risks]

Technical Assessment Workflow

Time Required: 4-6 hours
Audience: Technical architects, senior developers
Goal: Deep technical analysis and conversion strategy

Step 1: Data Architecture Analysis

- **DB Connections Sheet**: Catalog all connection types and infrastructure
- **Published Data Sources Sheet**: Map shared data dependencies
- **Embedded Data Sources Sheet**: Identify decentralized data patterns
- **Dataset Definitions Sheet**: Understand table structures and schemas
- Create data architecture diagram

Step 2: Custom Logic Inventory

- **Calculated Fields Sheet**:
    - Export full list with formulas
    - Categorize by complexity if possible (simple arithmetic → advanced analytics)

    - Identify Tableau-specific functions requiring conversion
    - Flag calculated fields used across multiple workbooks
- **Parameters Sheet**: Document interactive elements and user controls

Step 3: Visualization Complexity Assessment

- **Chart Types Sheet**: Analyze visualization distribution
- **Charts Report Sheet**: Deep dive into conversion challenges
- Group chart types by conversion difficulty
- Identify custom or advanced chart types requiring special handling

Step 4: Data Model Analysis

- **Table Relations Sheet**: Map all join relationships
- **Blended Relationships Sheet**: Document cross-source data blending
- Assess data model complexity impact on target platform
- Plan data preparation or ETL changes needed

Step 5: Platform-Specific Conversion Planning

- Map Tableau functions to target platform equivalents
- Document conversion patterns for common calculated field types
- Plan testing strategy for complex business logic
- Identify components requiring custom development

Technical Architecture Document:

- Data Source Migration Strategy: [infrastructure and connectivity]
- Custom Logic Conversion Approach: [calculated fields and parameters]
- Visualization Conversion Strategy: [supported vs. custom approaches]
- Data Model Changes Required: [ETL or data preparation needs]
- Testing and Validation Plan: [business logic verification strategy]

Dependency Analysis Workflow

Time Required: 1-2 hours
Audience: Data governance teams, business analysts
Goal: Understanding data and business logic dependencies

Step 1: Data Source Dependencies

- **Published Data Sources Sheet**: List all shared data connections
- **DB Connections Sheet**: Map to underlying systems
- For each published data source, identify dependent workbooks
- Create impact analysis: [Data Source] → [Affected Workbooks] → [Business Impact]

Step 2: Cross-Workbook Dependencies

- **Calculated Fields Sheet**: Find similar formulas across workbooks
- **Parameters Sheet**: Identify common parameter patterns
- **Chart Types Sheet**: Group dashboards using similar visualizations
- Map opportunities for standardization

Step 3: Business Logic Dependencies

- **Calculated Fields Sheet**: Extract business rules and formulas
- Group by business function (Sales, Finance, Operations, etc.)
- Identify calculated fields that implement the same business logic
- Flag critical business rules that must be preserved exactly

Step 4: User and Access Dependencies

- **Workbooks Sheet**: Group by `owner_name` for stakeholder mapping
- **Dashboards Sheet**: Cross-reference with business user groups
- Map business-critical dashboards that cannot have downtime
- Identify training needs for consolidated dashboards

Step 5: Technical Dependencies

- **Blended Relationships Sheet**: Document complex data combinations
- **Table Relations Sheet**: Map data model dependencies
- **Parameters Sheet**: Identify interactive features users rely on
- Create technical dependency chain analysis

Dependency Analysis Report:

- Critical Data Sources: [list with business impact assessment]
- Shared Business Logic: [common calculated fields across workbooks]
- Cross-Workbook Standards: [opportunities for consolidation]
- User Impact Assessment: [training and change management needs]
- Technical Risks: [complex dependencies that could cause migration delays]

Quick Reference: Sheet Navigation by Workflow

Executive Review: Summary → Conversion Summary → Rationalization Summary → Rationalized Dashboards

Migration Planning: Summary → Workbooks → Dashboards → Conversion Summary → Charts Report → Rationalization Summary

Technical Assessment: DB Connections → Data Sources → Calculated Fields → Chart Types → Table Relations → Blended Relationships

Dependency Analysis: Published Data Sources → Calculated Fields → Parameters → Workbooks → Charts Report

Troubleshooting Common Issues

Issue: Empty Published Data Sources Sheet

- Solution: Environment uses primarily embedded data sources
- Impact: Less centralized governance, more workbook-specific dependencies
- Action: Focus analysis on Embedded Data Sources sheet

Issue: Very High Manual Conversion Required

- Solution: Consider **Wavicle's EzConvertBI Converter** whiteglove service. Alternatively, consider dashboard redesign vs. direct conversion
- Impact: Higher migration effort and timeline
- Action: Evaluate business value vs. conversion cost

Issue: Many "Not Supported" Chart Types

- Solution: Plan alternative visualization approaches as listed in chart types sheet
- Impact: User experience changes, training requirements
- Action: Prototype alternatives early in migration process

Issue: Complex Blended Relationships

- Solution: Consider **Wavicle's EzConvertBI Converter** whiteglove service. Alternatively, consider ETL-based data preparation
- Impact: Architecture changes beyond BI tool migration
- Action: Involve data engineering team in planning