The Information contained in this presentation is provided for informational purposes only.

While efforts were made to verify the completeness and accuracy of information contained in this presentation, it is provided “as is”, without warranty of any kind, express or implied.

Canrock Solutions will not be responsible for any damages arising out of the use of, or otherwise related to, this presentation or any other documentation.

Nothing contained in this presentation is intended to, or shall have the effect of creating any warranty or representation from Canrock Solutions or its affiliates.
Implementation Methodology

• Requirements
• PM: Waterfall, Agile ...
• Development Standards
• Sustainability, Scalability
• Layered approach
# USE METHODOLOGY ... THAT MAKES SENSE

## IMPLEMENTATION METHODOLOGY

### Stage | Activity | Deliverable
---|---|---
**Analyze** | • Set clear objectives and success criteria  
• Analyze requirements  
• Plan resources, effort, and timelines | • Requirements Document  
• Project Plan

**Design** | • Prototype key design challenges. Design alternatives. | • Solution Architecture  
  - Application Data Flow and Dependencies  
  - Dimensional Model and Cube-Dimensional matrix  
  - Data Integration  
  - Key business rules and processes  
  - Reporting strategy  
  - Security  
  - Technical infrastructure

**Configure** | • Set milestone for each 2-3 weeks period  
• Implement, unit test and review  
• Feedback loop to design | • Complete functional application module at the end of each milestone.

**Deploy** | • User acceptance testing (UAT)  
• End user training  
• Promotion to production | • Operational Guide  
• Training plan  
• Test (UAT) plan

**Operate** | • On-going support  
• Collect feedback | • Lessons learned and go forward plan
Simpler Solutions for Complex Requirements

Common Standards

- Naming conventions
- Development standards
- Use element names, not aliases
- Externalize parameters and assumptions
IMPLEMENTATION METHODOLOGY

LAYERED DESIGN, DRY PRINCIPLE

User Interface
- Input Templates
- Standard Reports
- Ad-hoc Analysis

Application Logic: Modules
- SALES
- CAPEX
- HR
- SG&A

Application Configuration
- Versions
- Time
- Global Parameters
- Dimensions
- Subsets
- Process Control

Low Level Utilities & Templates
- Subsets Handling
- Views Handling
- Attributes Handling
- Process Templates

- Data Zero Out
- Data Source initialization...

Security
Error Handling
Logging
Navigation

Shared Services

TM1

Copyright © 2011 Canrock Solutions, Inc. All rights reserved.
TM1 APPLICATION DESIGN CONSIDERATIONS

Dimensional Model

- Attribute, Hierarchy, Dimension?
- Measure Dimensions
- Time: Continuous, Discreet?
- Comments Implementation
  - Relevant Intersections
  - How many Cubes?

Implementation Methodology

- Requirements
- PM: Waterfall, Agile ...
- Development Standards
- Sustainability, Scalability
- Layered approach

TM1 Application
"ATTRIBUTE vs. HIERARCHY vs. DIMENSION" DECISION TREE

**Identify Data Entities**

- **Attribute**
  - Do they describe data or other elements? **yes**
  - Relationship with elements they describe? **1.. 0, 1, ∞**
  - Aggregation required? **no**
  - Organized Hierarchically? **no**
  - Do they define picklist values? **no**
  - **Attribute**

- **Hierarchy**
  - Alternate consolidations trees? **yes**
  - **Hierarchy**

- **Dimension**
  - **Dimension**

---

Copyright © 2011 Canrock Solutions, Inc. All rights reserved.
HOW MANY CUBES?

Key Considerations

- Distinct Business Process
- Distinct User Group
- Different Dimensionality
- Different Data Granularity

Example
- Capex vs. Sales
- Strategic Plan vs. Rolling Forecast
- Buyer vs. Store Manager
- Dimension Measure Matrix
- Product Category vs. Product

Other Considerations

- Usability
  - Lookup vs. Input vs. Assumption vs. Reporting
- Data Redundancy
  - Load same actuals to multiple cubes
- Performance
  - Read vs. Write Inter-Cube rules & data movement
- Security
  - Comments cube
- Reporting
  - Cube vs. Element Security (Payroll)
  - More than 1 cube in a single report
  - Ad-hoc analysis

Less Objects => Less Maintenance
TM1 APPLICATION DESIGN CONSIDERATIONS

Dimensional Model

- Attribute, Hierarchy, Dimension?
  - Measure Dimensions
- Time: Continuous, Discreet?
- Comments Implementation
  - Relevant Intersections
- How many Cubes?

Data Integration

- List all Data Sources
  - Access: ODBC, CSV?
- Manual data sources
- No Datawarehouse?
- Data Source format?
- ETL: TI, ETL Tools...?
  - Update frequency?
  - Data growth

Implementation Methodology

- Requirements
- PM: Waterfall, Agile ...
- Development Standards
- Sustainability, Scalability
- Layered approach

TM1 Application

Copyright © 2011 Canrock Solutions, Inc. All rights reserved.
Separate ETL from Application Logic.

Standard Source Format.

<table>
<thead>
<tr>
<th>Metadata</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elem code</td>
<td>Dim 1 code</td>
</tr>
<tr>
<td>Parent code</td>
<td>Dim 2 code</td>
</tr>
<tr>
<td>Elem name</td>
<td>...</td>
</tr>
<tr>
<td>Elem type</td>
<td>Dim N code</td>
</tr>
<tr>
<td>Elem weight</td>
<td>Measure code</td>
</tr>
<tr>
<td>Elem format</td>
<td>Value</td>
</tr>
<tr>
<td>Attributes ...</td>
<td></td>
</tr>
</tbody>
</table>

Copyright © 2011 Canrock Solutions, Inc. All rights reserved.
Parameterized Query

Set Control Parameters

Write Query

Select ... where month between ‘%sStartPrd%’ and ‘%sEndPrd%’

Expand Query

sSQL = DataSourceQuery;
sStartPrd = CellGetS('ADMIN_CONTROL','Start Period','SValue');
sEndPrd = CellGetS('ADMIN_CONTROL','End Period','SValue');

DataSourceQuery = Expand(sSQL);

Zero-Out Utility

Source (Audit) Dimension
**TM1 APPLICATION DESIGN CONSIDERATIONS**

**Dimensional Model**
- Attribute, Hierarchy, Dimension?
- Measure Dimensions
- Time: Continuous, Discreet?
- Comments Implementation
- Relevant Intersections
- **How many Cubes?**

**Data Integration**
- List all Data Sources
- Access: ODBC, CSV?
- Manual data sources
- **No Datawarehouse?**
- Data Source format?
- ETL: TI, ETL Tools...?
- Update frequency?
- Data growth

**Implementation Methodology**
- • Requirements
- • PM: Waterfall, Agile ...
- • Development Standards
- • Sustainability, Scalability
- • **Layered approach**

**Business Logic**
- Consolidation, Data signs
- Rule or TI?
- Feeder Approaches
- Use of Subsets, Attributes
- Data Validation: Picklists
- Version Management
- What if

**TM1 Application**
RULE OR TI?

Legend:
- Complexity
- Dynamic Nature
- Sustainability
- Performance

Legend:
- Use Consolidation (+/-) if possible

C-level calculation
Usability Traceability
Real-time results

Deep Nesting (IF, Recursion)
Strings Calculations
Additional Calc Cubes required

Memory Consumption

Inter-cube Dependencies
Design Consistency

Cube Load, Query & TI time

Susceptibility to Metadata changes

Loop (While... End)
Complex Scope, Use Subsets
Complex Precedence, Validation

Seed Data
Seed Data

Sum Dynamic Ranges
Overfeeding

Freeze Results

Close to Real-time Action Button

Inter-cube Dependencies
Design Consistency

Copyright©2011 Canrock Solutions, Inc. All rights reserved.
Rule Best Practices (incremental development and testing)

**Development Standards**
- # Comments
  - # Indentation
- Use {} for common rules ('Electronics', 'Furniture')
- Do not hardcode, use element name, not Alias
- Use short reference [ ] vs. DB()
- Uniqueness ['country': 'us'][69]

**Rules Order**
- Organize rules from narrow to general scope
  - ['beverages', 'price'] = N: ...
  - ['price'] = N: ...

**Relationship Functions**
- Consider STET function on a separate line
  - ['Actual'] = N: STET;
- Instead of IF and repeating
  - [...] = N: IF (version @= 'Actual', STET, ...);

**Functions**
- Use Reference Guide
  - Consider new 9.5.2 Consolidation Functions:
    - ConsolidatedMin()
    - ConsolidatedMax()
    - ConsolidatedAvg()
    - ConsolidatedCount()
    - ConsolidatedCountUnique()

**Feeders**
- Use them. Check them.
  - Select feeders carefully
  - Avoid under-feeding, over-feeding
  - Refrain from conditional feeders if possible

---

TI Best Practices (incremental development and testing)

**Development Standards**
- # Comments
  - # Indentation
- Variables Convention
- Centralize Data Source initialization
- Declare user variables in the Prolog, Cleanup in Epilog
- Logging, Debugging TextOutput()

**Error Handling**
- Subset
  - SubsetExists (DimName, SubsetName)
- Element
  - DimIx (DimName, ElemName) = 0
- Child Process
  - sPrName = 'sub_create';
  - nPrReturn = ExecuteProcess(sPrName);
  - <<Error Handling>>
  - Skip to Epilog ProcessBreak;

**Tabs**
- Use Metadata tab for metadata updates
  - Use Data tab for attribute values and data updates

**Sustainability**
- Layered approach
  - - Create Subset
  - - Create View
  - - Zero Out Data
  - Avoid repeating statements
  - Use While...END and Expand
    - s1 = 'Core'; s2 = 'Non-Core';
    - nPos = 2;
    - WHILE (nCnt <= nPos);
    - sElem = Expand ('%'s'
      | NumberToString(nCnt)
      | '%');
    - END;

**Performance**
- Move Data preparation to ETL layer
  - Only include required values in source view
  - Separate data and metadata updates
  - Use TM1Top() to fine-tune your processes

---

Copyright © 2011 Canrock Solutions, Inc. All rights reserved.
**TM1 Application Design Considerations**

**Dimensional Model**
- Attribute, Hierarchy, Dimension?
- Measure Dimensions
- Time: Continuous, Discrete?
- Comments Implementation
- Relevant Intersections
- How many Cubes?

**Data Integration**
- List all Data Sources
- Access: ODBC, CSV?
- Manual data sources
- No Datawarehouse?
- Data Source format?
- ETL: TI, ETL Tools...?
- Update frequency?
- Data growth

**Implementation Methodology**
- Requirements
- PM: Waterfall, Agile ...
- Development Standards
- Sustainability, Scalability
- Layered approach

**TM1 Application**
- Business Logic
- User Interface

**Tools Selection**
- Rule or TI?
- Active Form or Slice?

**User Defaults**
- Use of Subsets, Attributes
- Feeder Approaches
- Data Validation: Picklists
- Version Management: What if

**Data Integration**
- Consolidation, Data signs

**User Interface**
- Complex Report Layout
- Report Distribution

Copyright © 2011 Canrock Solutions, Inc. All rights reserved.
<table>
<thead>
<tr>
<th>Perspectives</th>
<th>TM1 Web</th>
<th>Contributor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Purpose</td>
<td>• Manage TM1 applications</td>
<td>Managed contribution</td>
</tr>
<tr>
<td></td>
<td>• Data Entry and Reporting</td>
<td>• Data Entry and Reporting</td>
</tr>
<tr>
<td></td>
<td>• Limited Admin Tasks</td>
<td>• Contributor Administrator</td>
</tr>
<tr>
<td>Technology</td>
<td>Excel add-in</td>
<td>ASP.Net Web application</td>
</tr>
<tr>
<td>Data Entry</td>
<td></td>
<td>Java Web application</td>
</tr>
<tr>
<td>Slice and Dice, Pivot</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Charting</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Report / Template Layout Flexibility</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Customization</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Action Buttons Support</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picklist, Sandbox</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Modeling</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Built-in Workflow</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Security</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Purpose</td>
<td>Perspectives</td>
<td>TM1 Web</td>
</tr>
<tr>
<td>------------------------------</td>
<td>--------------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td>• Manage TM1 applications</td>
<td>• Data Entry and Reporting</td>
</tr>
<tr>
<td></td>
<td>• Data Entry and Reporting</td>
<td>• Limited Admin Tasks</td>
</tr>
<tr>
<td></td>
<td>• Create and publish TM1 websheets</td>
<td></td>
</tr>
<tr>
<td>Technology</td>
<td>Excel add-in</td>
<td>ASP.Net Web application</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data Entry</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Slice and Dice, Pivot</td>
<td>Through Cube Viewer</td>
<td>Through Cube Viewer</td>
</tr>
<tr>
<td>Charting</td>
<td>Full Excel based charting capability</td>
<td>Limited support</td>
</tr>
<tr>
<td>Report / Template Layout Flexibility</td>
<td>Complete flexibility: 2 modes: active form and cell-based slice</td>
<td>Excel functionality not 100% equal.</td>
</tr>
<tr>
<td>Customization</td>
<td>Complete VBA access</td>
<td>Limited through configuration</td>
</tr>
<tr>
<td>Action Buttons Support</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Picklist, Sandbox</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Modeling</td>
<td>Yes through Server Explorer</td>
<td>No</td>
</tr>
<tr>
<td>Built-in Workflow</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Security</td>
<td>Any mode</td>
<td>Any mode</td>
</tr>
</tbody>
</table>
**User Default**

- Create attribute on }Clients dimension
- Set Attribute Value to default element
- Use function TM1User() to get active user name
- Use DBRA() function to get default element for active user
- Pass default element to SubNm() function

**Default Element**

```
default_elem
Company = C1360
Server = canrock
TM1User ("canrock") = alex
```

**Dynamic Reports (DEMO)**

- **Link Title elements to TM1RptRow definition using**
  - Subsets
  - MDX
- **Select Current Element**
  - SUBNM() with DBRA()
- **Dynamic Zero Rows suppression**
- **Top (N) count based on Measure X**

- **Reference Names not Aliases in DBRW()**
- **Attribute based Conditional Formatting**
- **Excel based Running Total**
SUMMARY – DO IT THE RIGHT WAY!

Strong standards.
Iterate: focus on working modules.

Prototype Business rules.
Select best option (Rule/TI).

Sustainability
Scalability

Match User Interface with user needs.
Design dynamic reports.

Layered approach to model design.
Dimensional Model is Key.
Analyze alternatives.

Data Integration Framework in Phase 1.
Separate Application logic from ETL.
[ ] = ‘Q’ | ‘A’ ;