
ARM vs Excel for Large Models

August 2003



How is ARM better than Excel for large, complex models?

ARM is better than Excel for large models in three major ways

- ARM's Architecture
- ARM's Usability
- ARM's Configurability



Architecture



ARM's architecture provides major advantages

- Multi-user
- Secure
- Scalable
- Available over the web



Multi-user

- Multiple people can access the ARM information concurrently
- Each person gets his / her own view, independent of others
- For each user, ARM remembers
 - the windows last viewed
 - the timeframe of interest
 - the viewing frequency (distinct from the stored frequency)



Secure

- Security is managed at multiple levels

- Each user has a security profile
 - Security defines which objects a user can view
 - Security can also be defined for each individual measurement

- Users can have any or all of the following abilities:
 - Ability to administer security
 - Ability to configure / change the model
 - Basic user privileges



Scalable

- Scalability is provided in two dimensions
 - Size
 - An ARM model has essentially no size limit
 - Performance
 - The ARM database can run on anything from small Windows servers to very large Unix-based machines
 - For very large models, the database can be split across several machines to allow for unlimited growth

- ARM will continue to effectively run your application as your model grows



Available over the web

- ARM runs over the Internet in three ways
 1. In a browser (using a thin java client)
 2. Started in a browser, but running over IP with windows not restricted to the browser (using a thin java client)
 3. Using a thick client over IP

- Each option has its advantages and disadvantages depending on how you wish to deploy ARM



ARM Architecture – Summary

- As demonstrated, ARM’s architecture provides major advantages over Excel by being
 1. Multi-user
 2. Secure
 3. Scalable
 4. Available over the web

- In contrast
 1. Excel allows for a single user at a time
 2. Excel only provides security at the workbook level
 3. Excel has a fixed maximum size, and slows down as it grows
 4. Excel is available over the web, but is cumbersome to use as you must wait for the file to download fully first



Usability



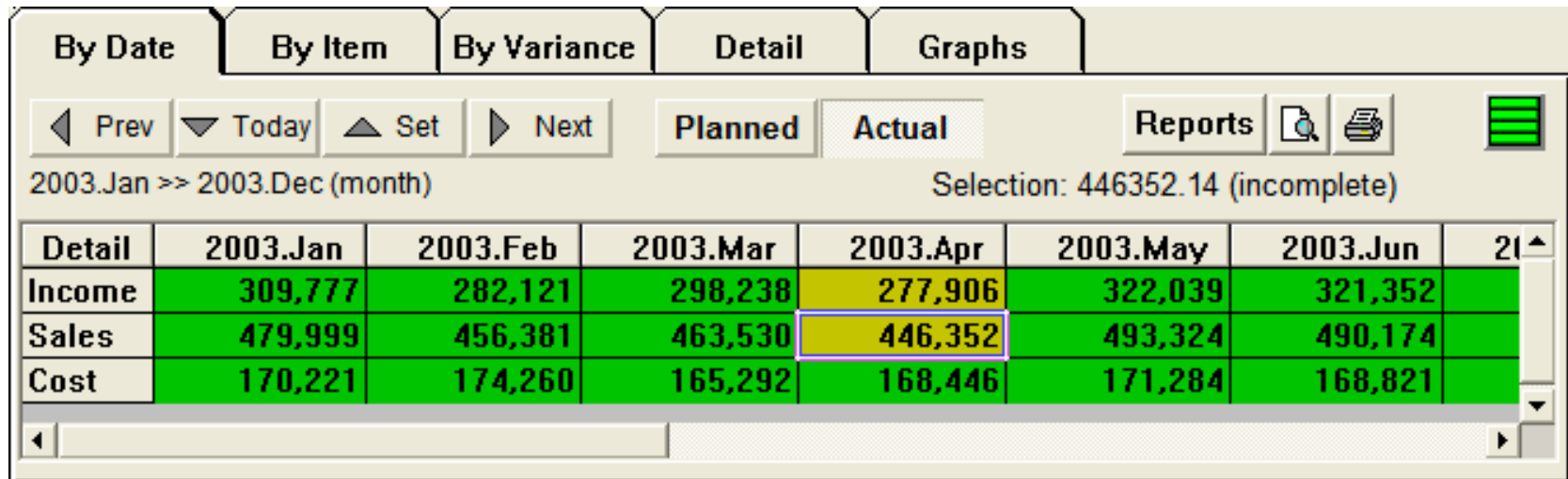
ARM is much easier to use than Excel on large models

- A hierarchy of data with full drill down capability
- Multiple ways to navigate
- Customized information views for each user
- Can view any information over time and across dimensions
- Context for each measure
- Reporting capability using Crystal Reports or FormatX



A hierarchy of data with full drill down capability

- The source and intermediate calculations for all values and statuses are easy to see via the “drilldown” feature of ARM
 - All “good” statuses show in green, so the user can easily see which values require attention
 - In the diagram below, Sales (\$446,352) for April 2003 is in yellow, indicating something is wrong



The screenshot shows the ARM software interface. At the top, there are tabs for 'By Date', 'By Item', 'By Variance', 'Detail', and 'Graphs'. Below these are navigation buttons: 'Prev', 'Today', 'Set', and 'Next'. There are also buttons for 'Planned' and 'Actual', and a 'Reports' button with a magnifying glass icon. The date range is '2003.Jan >> 2003.Dec (month)' and the selection is '446352.14 (incomplete)'. The main table displays financial data for 2003, with columns for months from Jan to Jun and a partial column for 2003. The rows are Income, Sales, and Cost. The Sales value for April 2003 is highlighted in yellow, indicating a problem.

Detail	2003.Jan	2003.Feb	2003.Mar	2003.Apr	2003.May	2003.Jun	2003.Jul
Income	309,777	282,121	298,238	277,906	322,039	321,352	
Sales	479,999	456,381	463,530	446,352	493,324	490,174	
Cost	170,221	174,260	165,292	168,446	171,284	168,821	



Full drill down capability (cont'd)

- Following the drill down below:
 - \$446,352 is made up of three values (257,886 141,280 and 47,185), with the first of these values “incomplete”
 - \$257,886 is also made up of three values, with the first one “incomplete”
 - Here we find the culprit. \$190,019 is made up of two values, the second of which is null (not entered)

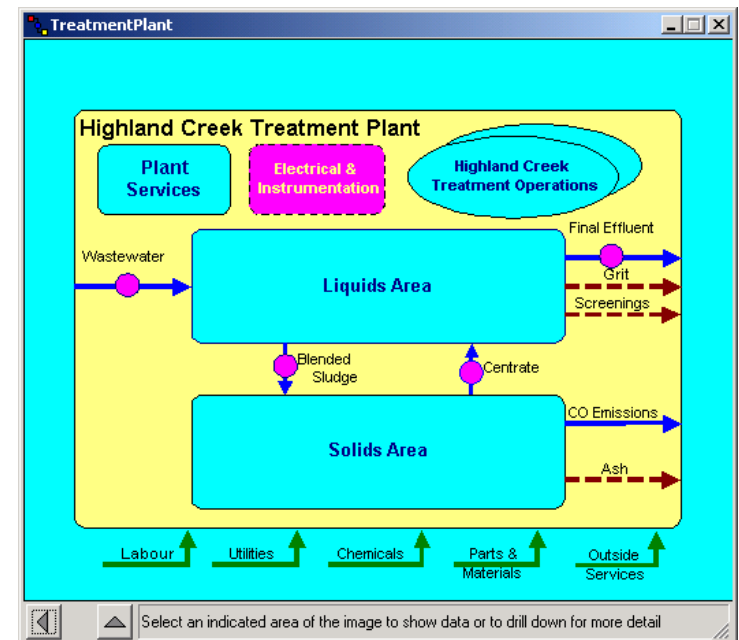
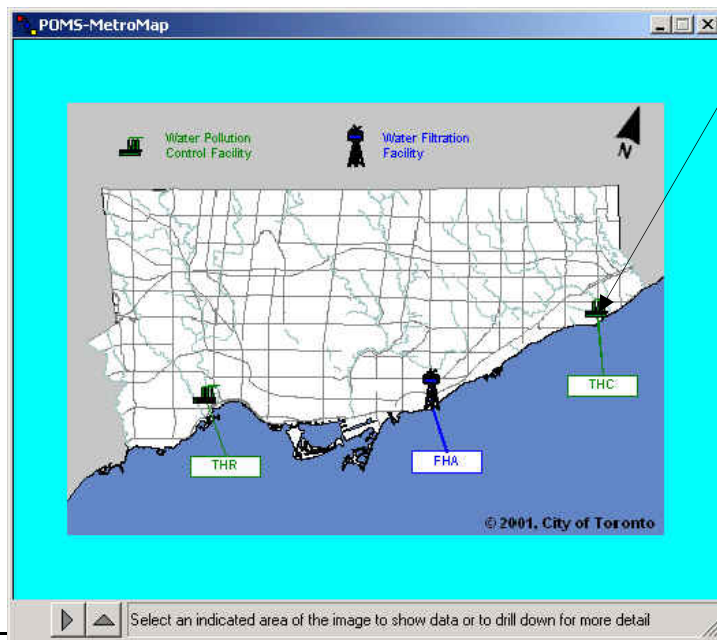
Item	Plan Date	Value	Function	Status
[-] Corp:ABC Co.>2003> Sales:' Sales Received' (\$)	2003.Apr	€ 446,352	sum	incomplete
[-] Corp Department:ABC HW>2003> Sales-Dept:' Sales Received' (\$)	2003.Apr	€ 257,886	sum	incomplete
[-] Province Department:ON HW>2003> Sales-Dept:' Sales Received'	2003.Apr	€ 190,019	sum	incomplete
[-] Store Department:TO:101 HW>2003> Sales-Dept:' Sales Receive	2003.Apr	€ 190,019		good
[-] Store Department:HM:102 HW>2003> Sales-Dept:' Sales Receiv	2003.Apr	****		null
[+] Province Department:PQ HW>2003> Sales-Dept:' Sales Received'	2003.Apr	€ 52,715	sum	good
[+] Province Department:BC HW>2003> Sales-Dept:' Sales Received'	2003.Apr	€ 15,151	sum	good
[+] Corp Department:ABC SG>2003> Sales-Dept:' Sales Received' (\$)	2003.Apr	€ 141,280	sum	good
[+] Corp Department:ABC LW>2003> Sales-Dept:' Sales Received' (\$)	2003.Apr	€ 47,185	sum	good



Multiple ways to navigate – Graphical

- Graphical navigation to information
 - Navigate to information using pictures (GIFs), with user-defined hot points and navigation paths
 - Best for most users

Click here to get map at right



Multiple ways to navigate – Explorer-like

■ Explorer-like navigation

- Navigate through the model using model components and their relationships
- Best for configurators and those more comfortable with the model

The screenshot shows a software window titled "Explore: Detail (2003 > ABC Co. [Corp]) (Item Worksheet)". The window has a menu bar (File, Options, Windows, Help) and a toolbar with buttons for "By Date", "By Item", "By Variance", "Detail", and "Graphs". Below the toolbar are navigation buttons: "Prev", "Today", "Set", "Next", "Planned", and "Actual". There are also "Reports" and "Print" icons. The main area displays a table for the period "2003.Jan >> 2003.Dec (month)".

Detail	2003.Jan	2003.Feb	2003.Mar	2003.Apr	2003.May	2003.Jun	2003.Jul	2003.Aug
Income	309,777	282,121	298,238	277,906	322,039	321,352	295,505	327,108
Sales	479,999	456,381	463,530	446,352	493,324	490,174	466,209	492,619
Cost	170,221	174,260	165,292	168,446	171,284	168,821	170,705	165,511

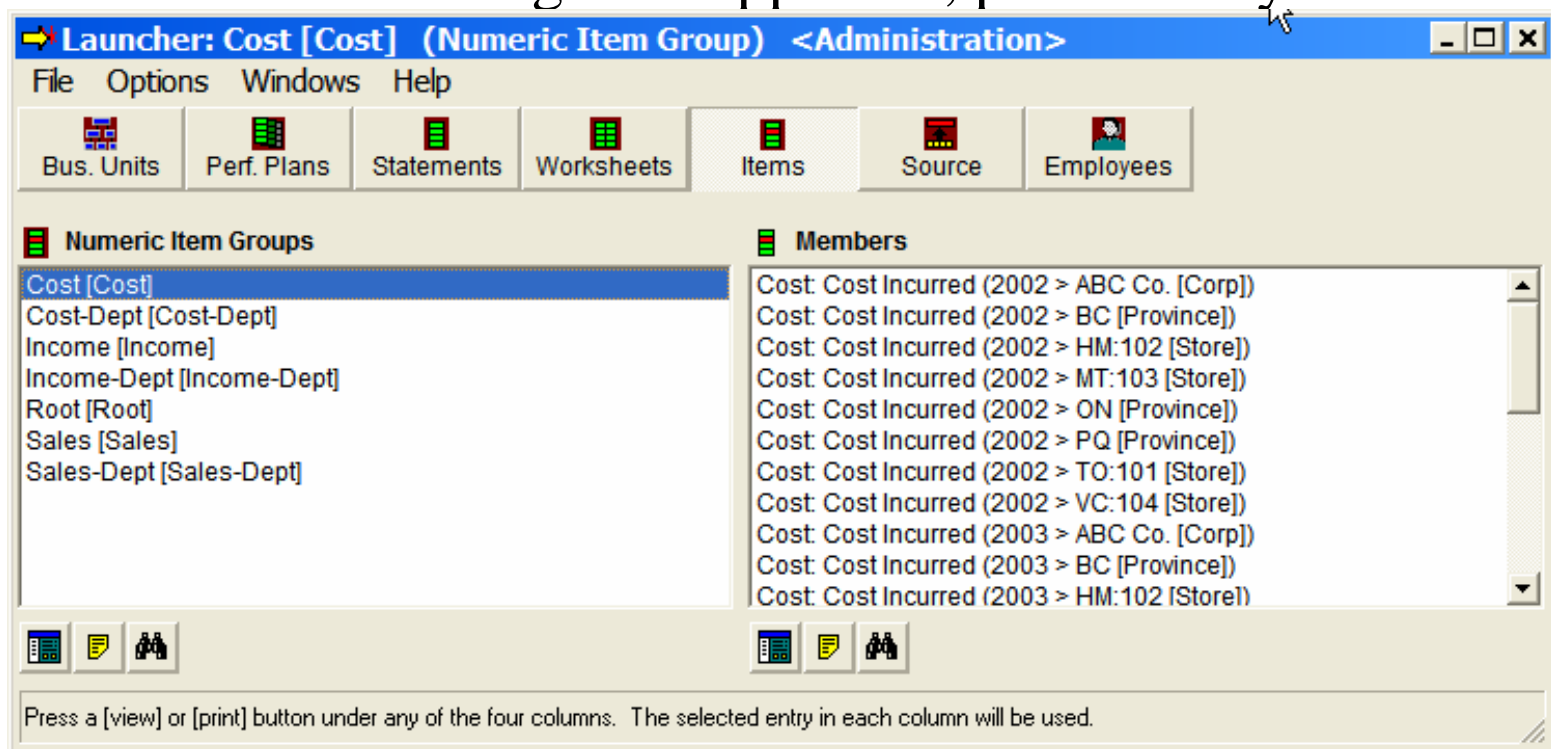
At the bottom of the window, there is a "Report" button, a "Use data view" checkbox, and "Save" and "Cancel" buttons. A status bar at the very bottom indicates: "Current selection is Item Worksheet: Detail (2003 > ABC Co. [Corp]). Permission level is UPDATE".



Multiple ways to navigate – By Object Type

■ Navigation by object type

- Navigate by model component definitions and their members
- Alternative navigation approach, preferred by some



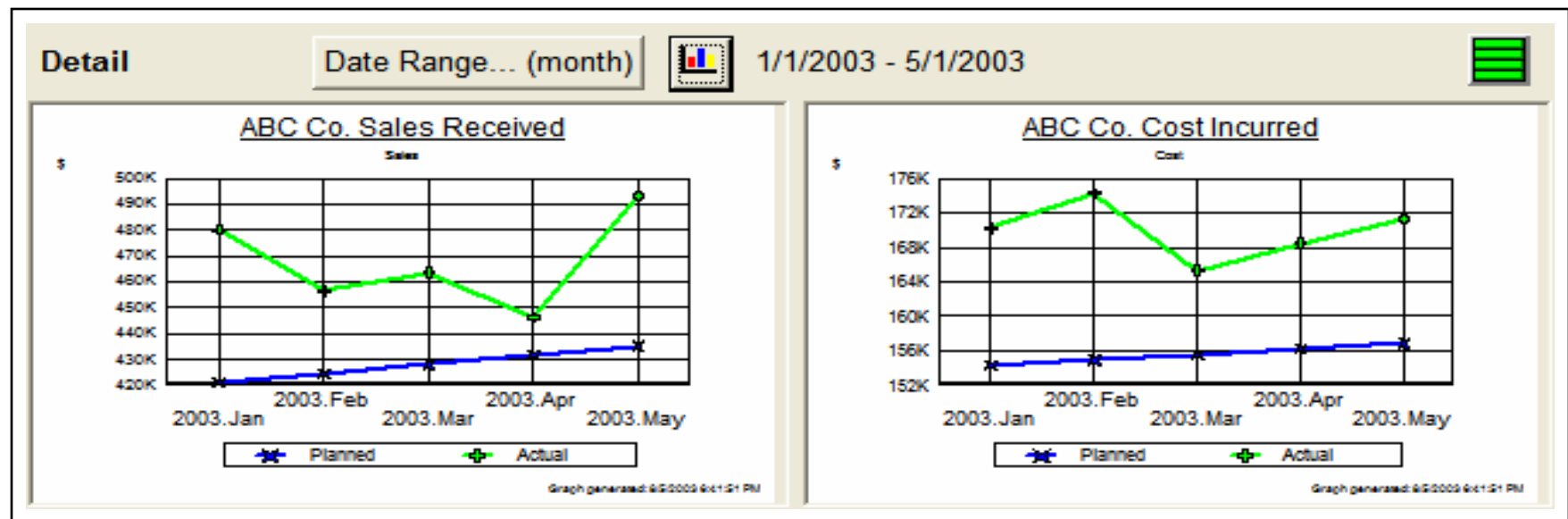
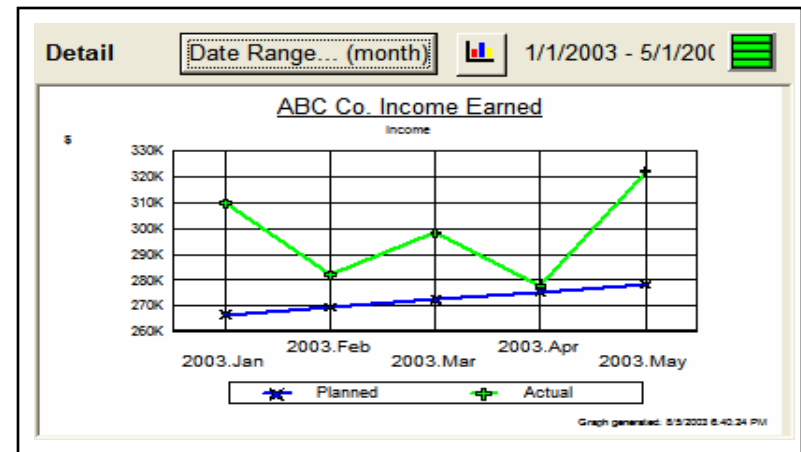
Customized information views for each user – Tabular vs Graphical (single or multiple)

By Date | By Item | By Variance | Detail | Graphs

◀ Prev | Today ▼ | ▲ Set | ▶ Next | Planned | Reports

2003.Jan >> 2003.May (month)

Detail	2003.Jan	2003.Feb	2003.Mar	2003.Apr	2003.May
Income	309,777	282,121	298,238	277,906	322,039
Sales	479,999	456,381	463,530	446,352	493,324
Cost	170,221	174,260	165,292	168,446	171,284



Customized information views for each user – Monthly vs Quarterly vs Yearly

Detail	2002.Jan	2002.Feb	2002.Mar	2002.Apr	2002.May	2002.Jun	▲
Income	287,500	286,328	291,489	304,790	302,534	312,054	
Sales	437,500	431,745	441,278	452,095	452,599	459,812	
Cost	150,000	145,417	149,790	147,305	150,065	147,758	

Detail	2002.Q1	2002.Q2	2002.Q3	2002.Q4	2003.Q1	2003.Q2	▲
Income	865,317	919,378	844,205	913,186	890,136	921,298	
Sales	1,310,523	1,364,505	1,316,309	1,418,043	1,399,910	1,429,850	
Cost	445,207	445,128	472,104	504,857	509,774	508,552	

Detail	2002	2003	Summary	▲
Income	3,542,085	3,867,055	7,409,140	
Sales	5,409,380	5,867,662	11,277,043	
Cost	1,867,295	2,000,608	3,867,903	



Customized information views for each user – Other Possibilities

- ARM remembers each user's view preferences, including:
 - Graphical vs tabular vs variances views
 - Date range selected (Jan 1 – Feb 15 vs. Dec 15 – Dec 18)
 - Measurement frequency (daily, weekly, monthly, etc.)
 - Variances view
 - e.g. view planned vs actual for a specific measurement
 - Graphical options
 - View some or all measures for the timeframe
 - View some or all dimensions (e.g. planned and actual) for any measurement

- ARM also provides for multiple tabular views to allow the user to see only those measures of interest for any object



Can view any information over time and across dimensions

- All views provide access to information over time

Non-Department View	2002.Nov	2002.Dec	2003.Jan	2003.Feb	2003.Mar	2003.Apr	2003.
Income	310,240	308,595	309,777	282,121	298,238	277,906	32
Sales	476,192	484,175	479,999	456,381	463,530	446,352	49
Cost	165,952	175,581	170,221	174,260	165,292	168,446	17

- All views provide access to variance information

Non-Department View [2003 > ABC Co. [Corp]]	Planned	Act. - Pla.	Actual	Variance comment
Income	293,541	56,969	350,510	Excellent Income, well above planned value
Sales	453,677	60,698	514,374	
Cost	160,135	3,728	163,864	Cost somewhat higher than planned, due to ...



Each measure has context

- Each measure is not just a number. Associated with each measure, ARM maintains:
 - A name, an ID and a description
 - The formula for calculating the value and the other measures that are inputs to that formula (e.g. sum)
 - The format of that number
 - Number of decimal places
 - Any prefix or suffix (e.g. “\$” or “hr”)
 - Which object it is measuring (e.g. Pump A)
 - An associated status, giving an indication of the validity of that number



Reporting capability – using Crystal Reports

- Reports can be customized using Crystal Reports

Report: Non-Department View On: Non-Department View In BU: ABC Co.

<u>Date</u>	<u>Income</u>	<u>Sales</u>	<u>Cost</u>
2002.Nov	310,239	476,191	165,952
2002.Dec	308,594	484,175	175,580
2003.Jan	309,777	479,998	170,221
2003.Feb	282,120	456,381	174,260
2003.Mar	298,238	463,530	165,292
2003.Apr	277,905	446,352	168,446
2003.May	322,039	493,323	171,284
2003.Jun	321,352	490,173	168,821
2003.Jul	295,504	466,209	170,704
2003.Aug	327,108	492,619	165,511

Page 1 of 1 Printed: 8/6/2003 12:54:10PM



Reporting capability – using FormatX

- All views support formatted output to Excel using FormatX

ABC Co

Non-Department View

Non-Department View	Income	Sales	Cost
2002.Nov	310,240	476,192	165,952
2002.Dec	308,595	484,175	175,581
2003.Jan	309,777	479,999	170,221
2003.Feb	282,121	456,381	174,260
2003.Mar	298,238	463,530	165,292
2003.Apr	277,906	446,352	168,446
2003.May	322,039	493,324	171,284
2003.Jun	321,352	490,174	168,821
2003.Jul	295,505	466,209	170,705
2003.Aug	327,108	492,619	165,511

8/6/2003 1:01 PM

Page 1 of 1



ARM Usability – Summary

- From a usability point of view, these features set ARM apart from Excel
 1. Full drill down capability
 2. Multiple ways to navigate
 3. Customized information views for each user
 4. Can view any information over time and across dimensions
 5. Context for each measure
 6. Reporting capability using Crystal Reports or FormatX



ARM Usability – Summary (cont'd)

■ Contrasting with Excel

1. In a complicated spreadsheet (or set of spreadsheets), even drilling down one level can be a challenge
2. If you don't know where data is located in a complicated set of Excel sheets, you are in for a long search
3. Excel has no concept of a user, and even if it did, there is nothing even approaching the flexibility of ARM with respect to views
4. Views over time and across dimensions are “free” with ARM, and very difficult to do with Excel
5. In Excel, each number is isolated. No grouping of numbers is supported
6. Reports are possible in Excel, but require much manual work



Configurability



ARM's configuration tools yield robust, manageable, auditable models to match your organization

- **Robust and manageable models** are achieved using reusable components
- Models can be **validated** by means of ARM's drill down capability
- Out of range numbers can be flagged or trigger e-mail notification
- ARM models support both unlimited depth and unlimited numbers of consolidations
- Parts of the model are provided with no effort



Models are based on reusable “components”

- Formulas are defined once and used as often as required
- Each object type knows its measures, calculations and views
 - Every time a new object is defined, ARM automatically generates the following based on its type:
 - a place to store all the measures over time and across dimensions
 - the appropriate calculations for that object
 - all the views for that object
 - Calculations in one object which depend on measures in other objects are also managed automatically



Models are based on reusable “components” (cont’d)

■ Example 1:

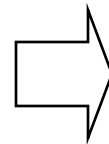
- If ‘Income’ is defined as ‘Revenue’ – ‘Cost’, every object which has the measures ‘Income’, ‘Revenue’ and ‘Cost’ automatically know and implement that relationship

Model-wide definition

‘Income’ = ‘Revenue’ – ‘Cost’

If the following objects are defined to have Income, Revenue and Cost:

- Store A
- Store B
- Product C



- Store A
 - Store B AND
 - Product C
- automatically calculate ‘Income’ by subtracting ‘Cost’ from ‘Revenue’



Models are based on reusable “components” (cont’d)

■ Example 2:

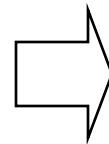
- If ‘Revenue’ is the sum of the ‘Revenue’ measures for all linked objects, any time a link is added to an object, the ‘Revenue’ from that linked object is automatically included

Model-wide definition

‘Revenue’ = Sum (‘Revenue’ of all linked objects)

Region A is linked to :

- Store A
- Store B
- Store C



The ‘Revenue’ of Region

A is the sum of the

‘Revenue’s in

- Store A
- Store B and
- Store C



Models are based on reusable “components” (cont’d)

■ Example 3:

- If an object definition includes a set of measures, any object with that object definition will get the same set of measures

If the ‘Region’ definition contains the measures:

- Revenue
- Cost
- Income

AND

10 Regions are made using the ‘Region’ definition:

i.e. Region01 ... Region10

1. Each Region defined will have all of the measures defined
2. The measures will be related to each other as described in example 1 and example 2



Models are based on reusable “components” (cont’d)

- By modeling with reusable components
 - Common sets of measures are defined in one place
 - Common functions are defined in one place
 - Common views are defined in one place
 - Functions which depend on values outside of the current object are defined in one place

- When the actual objects and their links are defined
 - Measures are automatically generated
 - Views are automatically generated
 - Calculations are automatically generated

- As a result, models generated in ARM are both robust and changeable



Models can be validated by means of ARM's drill down capability

- As described in the usability section, all values can be traced back to their sources
 - Certifying a model correct is painstaking, though doable

- In a previous ARM implementation, where the goal was to replace a very complicated set of Excel spreadsheets
 - Many discrepancies existed between the original Excel sheets and the ARM model generated
 - Many hours were spent “digging” through the spreadsheets to understand the source information
 - In the vast majority of cases, the discrepancy was found to be a problem with the Excel sheets
 - Essentially, the wrong tool was being used. Excel is not robust enough to use for a large model



Out of range numbers can be flagged or trigger e-mail notification

- If sales go below a certain value or if the reserve capacity of a reservoir goes below some warning value, you can tell ARM to automatically mark the value with a warning status
- If sales go below a very critical value or if the reservoir capacity goes below some critical value, you can tell ARM to automatically e-mail that fact to a defined set of users



ARM models support both unlimited depth and unlimited numbers of consolidations

- No limit to the number of levels of hierarchy
 - Your model can grow to match your organization

- No limit on the number of ways numbers can be consolidated
 - Numbers can be consolidated by geography, by product, by manager, by cost center, or any number of ways

- Excel works very well for two dimensional models, but becomes difficult for real-world situations which are more complicated



Parts of the model are automatic / free

- Simply by using an ARM model, the following features (among others) happen with no (or very little) effort by the configurator
 - Management of values over time
 - All values are stored at a base frequency (e.g. hourly, monthly), but can be viewed at that frequency or less frequent (e.g. daily values can be viewed weekly)
 - Maintaining statuses with numbers
 - All numbers have associated statuses, and those statuses are automatically used in calculations to highlight issues
 - Multiple dimensions
 - e.g. planned and actual (and any number of other parallel dimensions) are automatically supported



ARM Configurability – Summary

- From a configurability point of view, these features set ARM apart from Excel
 1. **Robust and manageable models** are achieved using reusable components
 2. Models can be **validated** by means of ARM's drill down capability
 3. Out of range numbers can be flagged or trigger e-mail notification
 4. ARM models support both unlimited depth and unlimited numbers of consolidations
 5. Parts of the model are provided with no effort



ARM Configurability – Summary (cont'd)

■ Contrasting with Excel

1. It is very difficult to get a large Excel model correct, and even more difficult to change that model once developed
2. Validating a large Excel model is extremely difficult. Cells can be dependent on other cells in very complicated ways that are very tricky to follow
3. Excel has the capability to flag numbers and e-mail, but the logistics required to manage such a capability would be prohibitive
4. Strictly speaking, Excel is essentially unlimited in size (given enough spreadsheets), but becomes unwieldy very quickly
5. Excel provides very little help for the modeler



Summary



How is ARM better than Excel for large, complex models?

ARM is better than Excel for large models in three major ways

- ARM's Architecture
- ARM's Usability
- ARM's Configurability

