

## The use of Integrated Morphometry Analysis with MetaXpress and Variables

### ABSTRACT

In the current version of MetaXpress if you wish to use Integrated Morphometry Analysis (IMA) in an analysis you must use a journal and add the journal to the database using Add Analysis to Database. There are two key issues in using IMA within an analysis journal in MetaXpress. These issues are similar to using IMA with variables.

If you use the Summary Log with IMA the problem comes when you have more than one measurement. MDCStore records the first of the measurements. Variables record the last of the measurements.

A similar situation occurs with the Object Log of IMA. If you use the Object Log with IMA MDCStore records the first of the object measurements and variables record the measurements of the last object.

This document describes methods to use to record all the measurements and objects from your IMA analysis.

### OVERVIEW

You can use Journals in MetaXpress to record output from much of the measurement tools in MetaXpress, in addition to the data recorded from the application modules. To use journals for this purpose you need to move the journals to the C:\Assay directory and in MetaXpress use the Screening menu's "Add Analysis to Database" command. Any data logged in those journals will be stored as measurements in your database which can later be accessed via the Review Plate Data window.

Some problems occur when you try to use this method of data storage with IMA. The MDCStore database will only store the first measurement in the IMA configuration – only the first parameter being measured if you are logging summary information or only the first object if you are logging individual object data. A similar problem occurs if you are using variables to store the information from IMA (which you can then log to the database); you only can retrieve the last measurement in the IMA configuration – the last parameter being measured in a summary, or the last object.

### LOGGING SUMMARY DATA

If you use the Summary Log with IMA the problem comes when you have more than one measurement. MDCStore records the first of the measurements while variables record the last of the measurements. Given the following example, where we have an IMA analysis with three measurements being made: Total Area, Shape Factor, and Total Gray Value.

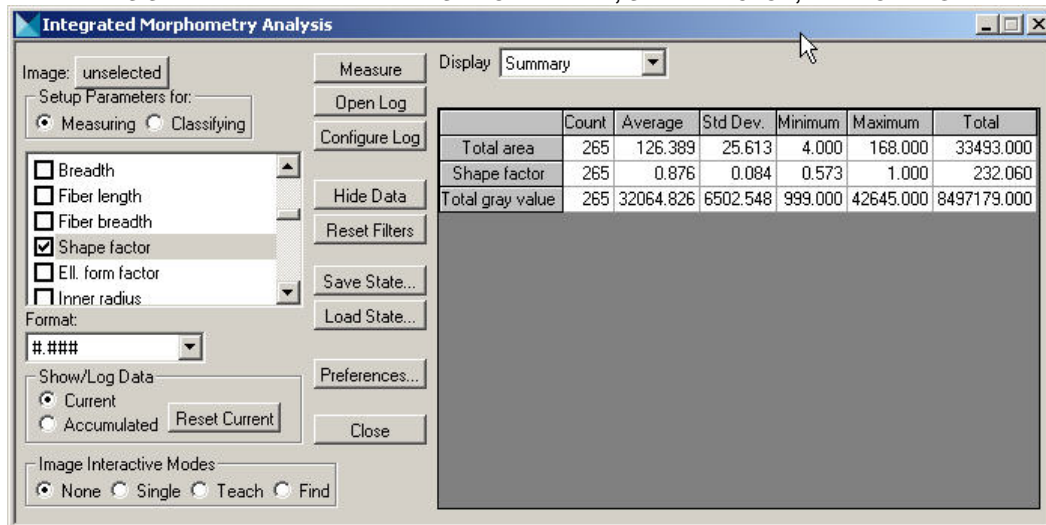
ARTICLE #  
T20054

PRODUCTS  
MetaXpress™

CREATED  
30-Mar-2006

LAST UPDATED  
30-Mar-2006

**FIGURE 1**  
**IMA ANALYSIS WITH THREE PARAMETERS – TOTAL AREA, SHAPE FACTOR, AND TOTAL GRAY VALUE**



When we run this analysis on our plates, we can review the data available on the Review Plate Data screen. As you can see from the next image, only the first parameter in the summary measurements is recorded. We can see Total Area, but not Shape Factor or Total Gray Value.

**FIGURE 2**  
**REVIEW PLATE DATA – ONLY RESULTS FOR TOTAL AREA VISIBLE**

	Parameter	Count	Average	Standard Deviation	Min	Max	Total	Header Label	Image Name	Image Plane
A01	Total area	9.00	23699.11	66803.60	5.00	212648.00	213292.00	Image Name	unselected	1.00
A02	Total area	265.00	126.39	25.61	4.00	168.00	33493.00	-	unselected	1.00
A03	Total area	13.00	15228.77	52304.17	2.00	196414.00	197974.00	-	unselected	1.00
A04	Total area	9.00	23699.11	66803.60	5.00	212648.00	213292.00	-	unselected	1.00
A05	Total area	265.00	126.39	25.61	4.00	168.00	33493.00	-	unselected	1.00
A06										
A07										
A08										
A09										
A10										

As an alternative, we could use Journals to record the values from IMA in the form of variables. When we do this, we get similar results, but instead of seeing Total Area, we only get Total Gray Value recorded in the database.

To get around this, we need to create several IMA state files, each only measuring one parameter. We will log summary data for each IMA state, and use variables to store the information we want to maintain. After doing all the measurements, we will log these variables to the database. A workflow for our journal, then, would be:

- Pause Logging to the Database (or pause summary logging)
- Load an IMA state that measures only one parameter (like Total Area)
- IMA – Measure
- IMA – Log Summary Data
- Assign Measurement Variables
  - Total\_Area\_Average = IMASummary.Average
  - And so forth
- Load an IMA state that measures just on other parameter (like Shape Factor)

- IMA – Measure
- IMA – Log Summary Data
- Assign Measurements Variables
  - Shape\_Factor\_Average = IMASummary.Average
  - And so forth
- Resume logging to the database (resume summary logging)
- Log the appropriate variables.

Once our journal is configured like this, we then have access to all the values from our individual IMA measurements, under the names we provide for the variable.

**FIGURE 3**  
**REVIEW PLATE DATA – ALL RESULTS FROM THE IMA ANALYSIS ARE AVAILABLE**

	Total_Area_Average	Total_Area_Total	Shape_Factor_Average	Shape_Factor_Total	Total_Intensity_Average	Total_Intensity_Total	Object_Count
A01	23699.11	213292.00	0.46	4.12	2338074.25	21042668.00	9.00
A02	126.39	33493.00	0.88	232.06	32064.83	8497179.00	265.00
A03	15228.77	197974.00	0.56	7.25	1561218.50	20295840.00	13.00
A04	23699.11	213292.00	0.46	4.12	2338074.25	21042668.00	9.00
A05	126.39	33493.00	0.88	232.06	32064.83	8497179.00	265.00
A06							
A07							
A08							
A09							
A10							
A11							

**FIGURE 4**  
**EXAMPLE JOURNAL USED TO GENERATE THE ABOVE DATA**

Measure Total Area, Shape Factor, and Total Grey Value to the database

Pause Logging and prepare the image

1: Pause Data Logging()  
 2: Pause Summary Logging()  
 3: Pause Object Logging()  
 4: Stop/Resume Logging to Database(Stop Logging)  
 5: Auto Threshold for Light Objects()

Measure Total Area, store Total Area variables

6: Integrated Morphometry - Load State("Total\_area")  
 7: Integrated Morphometry - Measure([Current At Start], -1)  
 8: Integrated Morphometry - Log Data([Current At Start], SUMMARY, CUF  
 Object\_Count = IMASummary.Count  
 Total\_Area\_Average = IMASummary.Average  
 Total\_Area\_Total = IMASummary.Total

Measure Shape Factor

9: Integrated Morphometry - Load State("Shape\_factor")  
 10: Integrated Morphometry - Measure([Current At Start], -1)  
 11: Integrated Morphometry - Log Data([Current At Start], SUMMARY, CL  
 Shape\_Factor\_Average = IMASummary.Average  
 Shape\_Factor\_Total = IMASummary.Total

Measure Total Gray Value

12: Integrated Morphometry - Load State("Total\_gray\_value")  
 13: Integrated Morphometry - Measure([Current At Start], -1)  
 14: Integrated Morphometry - Log Data([Current At Start], SUMMARY, CL  
 Total\_Intensity\_Average = IMASummary.Average  
 Total\_Intensity\_Total = IMASummary.Total

Resume Logging

15: Stop/Resume Logging to Database(Resume Logging)  
 16: Resume Data Logging()  
 17: Resume Summary Logging()  
 18: Resume Object Logging()

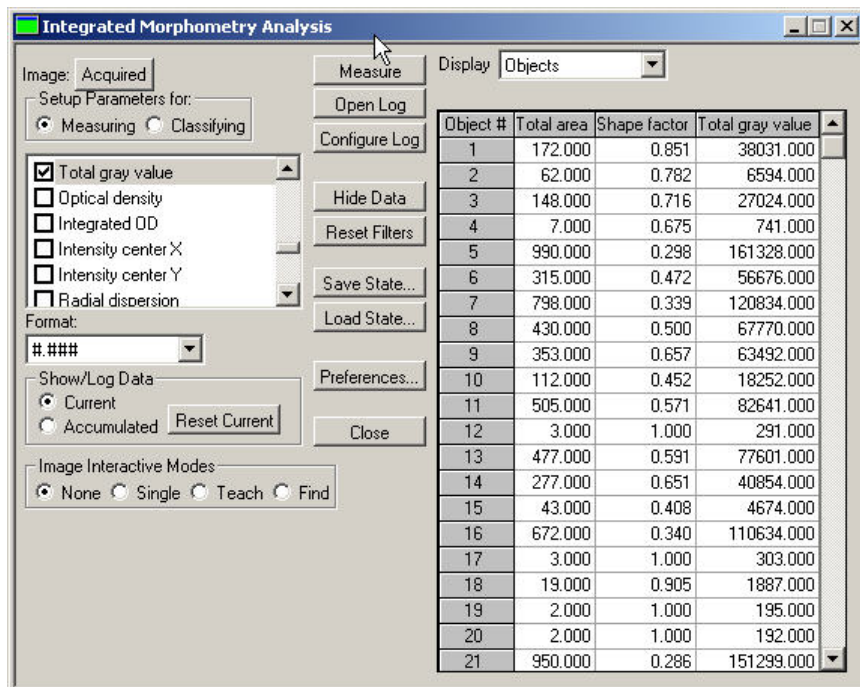
Log Our data from variables to the DB

19: Log Variable(Object\_Count, NEWLINE, HEADER)  
 20: Log Variable(Total\_Area\_Average, NEWLINE, HEADER)  
 21: Log Variable(Total\_Area\_Total, NEWLINE, HEADER)  
 22: Log Variable(Shape\_Factor\_Average, NEWLINE, HEADER)  
 23: Log Variable(Shape\_Factor\_Total, NEWLINE, HEADER)  
 24: Log Variable(Total\_Intensity\_Average, NEWLINE, HEADER)  
 25: Log Variable(Total\_Intensity\_Total, NEWLINE, HEADER)  
 ● \*\*\*\* End of Journal \*\*\*\*

**LOGGING OBJECT DATA**

A similar problem exists when trying to log individual object data using IMA. If you use Object Measurements in IMA, and have an analysis scheme that examines multiple objects, when you log that data to MDCStore, only the first object will be stored. This can be seen with the following example analysis:

**FIGURE 5  
 IMA ANALYSIS EXAMINING MULTIPLE OBJECTS IN ONE FIELD**



By using a journal configured to log the Object data with the above configuration, we can run an analysis on our acquired plates, and see the results on the Review Plate Data window. When we look at the data in the database, however, there is only one object logged – the first object on the image.

**FIGURE 6**  
**REVIEW PLATE DATA – ONLY FIRST OBJECT IN EACH WELL IS LOGGED**

	Object #	Total area	Shape factor	Total gray value
A01	1.00	212648.00	0.09	20975384.00
A02	1.00	111.00	0.81	28092.00
A03	1.00	196414.00	0.05	20102824.00
A04	1.00	212648.00	0.09	20975384.00
A05	1.00	111.00	0.81	28092.00
A06				
A07				
A08				
A09				
A10				
A11				

If we use variables to record our data then log those variables, only the last Object found in the image is logged. In order to solve this issue we need to create regions around our discreet objects and use IMA to measure each region individually. This is done by following this procedure:

- Set Preferences to not Measure All Regions
- Load an IMA state file that measures all your parameters of interest
- IMA Measure
- Create regions around objects
- Loop for all regions:
  - Assign a variable: Label = Region.Label
  - Set the Cell Number for database logging
    - Override the Cell Number with the Label variable created above
  - Perform an IMA Measure

- o IMA Log Object Data

Using a journal configured this way, you can log the information for each object in your image this way. The results will be displayed in your Cellular Results table, each object in the IMA analysis having its own row on the table.

**FIGURE 7**  
**CELLULAR RESULTS TABLE – EACH OBJECT FROM IMA ANALYSIS HAS ITS OWN ROW**

	Total area	Shape factor	Total gray value
1	196414	0.04934	2.01028e+007
2	82	0.520704	5550
3	349	0.68627	45852
4	5	0.806145	273
5	857	0.749344	121263
6	55	0.411745	4269
7	2	1	108
8	2	1	93
9	17	0.306868	1566
10	52	0.110706	5439
11	11	0.839954	651
12	56	0.475126	3453
13	72	0.289552	4500

**FIGURE 8**  
**OUTER JOURNAL – GENERATES REGIONS, CALLS THE INNER JOURNAL**

Measure IMA parameters for all objects and log them to the database

Make sure we measure only one region - and prepare

- 1: Preferences{}
  - 2: Clear Regions{[Current At Start]}
  - 3: Auto Threshold for Light Objects{}
- Generate our object mask
- 4: Integrated Morphometry - Load State{"ObjectM...
  - 5: Integrated Morphometry - Measure{[Current At S...
- Create regions around the objects, then measure and l
- 6: Create Regions Around Objects{[Current At Star...
  - 7: Loop for all Regions{[Current At Start], "IMA\_Dt...
- \*\*\*\* End of Journal \*\*\*\*

**FIGURE 9**  
**INNER JOURNAL DO IMA INDIVIDUALLY ON EACH OBJECT**

Measure IMA parameters for a single region and log the data to a Cell Number in the database

Record the current region label and set it as the curre

- Label = Region.Label
- 1: Set Cell Number for Database Logging{}
- ↳ Cell Number = val(Label)

Run IMA and log the data

- 2: Integrated Morphometry - Load State{"ObjectM...
  - 3: Integrated Morphometry - Measure{[Current At S...
  - 4: Integrated Morphometry - Log Data{[Current At S...
- \*\*\*\* End of Journal \*\*\*\*