# THE PROMETHION METABOLIC MEASUREMENT SYSTEM

**12/07/2013**

Table of Contents

# INTRODUCTION

The great physiologist Max Kleiber once memorably described the process of catabolism as “the fire of life” (#). This brings to mind the Greek legend of Prometheus, who gave the gift of fire to humankind. When Sable Systems International decided to build the best, most integrated metabolic and behavioral measurement system in the world, we needed to differentiate it from our other products. Remembering Kleiber’s beautiful metaphor, we derived the name of our new product line as a reminder of its primary purpose. Linguistically we might have called it “Prometheon”, but chose “Promethion” instead to minimize theistic overtones while providing the flame in our logo with a point of origination.

Four key features differentiate Promethion from other, superficially similar systems.

First, absolutely no “black box” operations take place. The system is completely transparent and traceable. Rather than a thin stream of data processed by mysterious and opaque algorithms, Promethion produces a deluge of raw data from which an immense variety of extracted data can be derived. Where needed for user feedback and real-time evaluation, it can calculate derived data (such as metabolic rate, food uptake and so on) on the fly, but the essence of the Promethion data stream is in its lack of any imposed analytical structure at the time of data acquisition. This lack of structure, paradoxically, makes the Promethion data stream far more powerful than the thin trickle of processed data emerging from legacy systems, because it imposes no limits or constraints on the questions you can pose to it and the answers you can obtain.

Second, the Promethion system does not impose binning restraints on the questions you can answer. All data from all sensors and other instruments in the system are acquired with a “heartbeat” of one Hertz. This offers unprecedented detail and versatility. Detail, because every nuance of every signal is captured; and versatility, because you are never hindered by pre-conceived operational or analytical paradigms hard-wired into the structure of the system.

Achieving the data density that is required to acquire free-form, raw data is an extremely difficult technical task, and required a radical re-design of existing instrumentation to achieve. In a typical 16-cage Promethion system, over 300 channels of data are acquired every second at extremely high resolution. Recording the resulting deluge of data only became practical quite recently, with the advent of storage media of ever-increasing capacity. Now, however, months or years of raw Promethion data can easily be stored on a wide variety of storage media at low cost and high convenience. Technological advances, and innovations in instrument communication architecture, have joined forces to make the unique Promethion approach possible.

Third, you may wonder how answers emerge from the data stream. The answer, which is also unique to Promethion, is: Analytical scripts, also called macros. They are easily understood, sequential operations that are performed on the raw data to yield the answers to the questions that your research asks. They are extremely powerful tools, not least because they can easily be changed to answer different questions, or answer the same questions in a different way, as your research evolves – without having to re-run expensive and time-consuming experiments.

The fourth area of Promethion uniqueness is a product of the first three. It was never even intended, but arose spontaneously from the unfettered structure of the data: Behavioral analysis. Because every interaction with every sensor by every animal is meticulously recorded, together with each animal's exact position at all times, it is possible to deduce a wide range of individual behaviors and place them in a temporal sequence. Moreover, the quantitative aspects of many behaviors are captured in a way that is simply impossible with more traditional means of behavioral monitoring, such as video analysis. As an example, mice often interact with the food hopper without actually removing any food from it. With Promethion's built-in behavioral analysis engine, such events are easily distinguished from actual food uptake and, in fact, constitute a separate behavior from food uptake. Time and ambulatory budgets, and transition probability matrices that provide a detailed breakdown of a mouse's mental operating system, are available in seconds. The result is a detailed, focal-animal mine of behavioral data that further enriches the variety of questions that can be asked of your Promethion system.

This manual will cover, in sequence, how the system works; setting up the system; and acquiring data from it. It will then cover data analysis in some detail.

Throughout, always remember that we at Sable Systems are dedicated to your success. If you have questions not addressed in this manual, or need clarification, we are here to help. You can contact us at 001-702-269-4446 or support@sablesys.com

.

# HOW THE SYSTEM WORKS

The Promethion system uses flow-through respirometry to monitor metabolic rates, and uses appropriate sensor technology to monitor masses (food and water uptake, body mass), position (X, Y and Z light gate arrays), wheel rotation and other variables. All measurements are optional; you can implement as many or as few as you need. You can use the same Promethion software to monitor a single food hopper, or a large array of cages complete with metabolic measurement and all available sensors. Unlike the competition, the Promethion system is quickly and easily re-configurable and expandable as your needs change, and there are no per-sensor charges or costly data aggregators that limit your flexibility.

As explained in the Introduction, all sensors, including all gas analyzers involved with metabolic measurement, are sampled once every second for the duration of each run. Sampling takes place through a high speed network that uses the same protocol as those found in jet aircraft and sports cars. This is called a *control area network* (or CAN bus). Thus all data transfer is entirely digital, and takes place at much higher resolution than is possible in analog-based systems. For example, Promethion mass sensors have a remarkable dynamic range of 1:500,000. The high speed of the CAN bus allows hundreds or even thousands of channels of unique, high-resolution data to be acquired every second with excellent reliability.

The Promethion CAN bus interfaces to your host computer using one of two different interface modules: The IM-1 (a USB-serial-CAN bus module, mostly used in earlier systems) or the IM-2 (a direct USB to CAN bus module, recommended for new installations). The Promethion software running on your host computer automatically recognizes which interface module is attached.

The CAN bus is simple to connect. To achieve maximum flexibility and expandability, the CAN bus is daisy-chained from one instrument or sensor to the next. Ordinary CAT-5 network cable is used, and is supplied with the system. For example, a small, 8-cage, multiplexed (see below) system will typically be daisy-chained in the order: interface, gas analyzer, flow generator, then CC-1 (cage controller), with one CC-1 per cage. You might wonder how the system distinguishes between the different CC-1s to which it is connected. It does so because each CC-1 has a unique address, corresponding to the cage number with which it is always associated. Each CC-1 is labeled with its address. Likewise with different analyzers or flow generators, if your system has more than one of each.

Each CC-1, in turn, uses its own, separate CAN bus to connect to the sensors (such as mass measurement modules (MM-1s) and beam break modules (BB-XYs or BB-Zs) associated with its cage. When a cage has multiple sensors of the same type, those sensors are differentiated by their addresses. Up to 100 different sensors can be connected to a single cage. Thus, each cage has its own network, with its attached CC-1 cage controller acting as that cage's gateway to the main network that links all of the instruments together.

This system gives you incredible versatility and expandability, but it's easy to use. You don't have to understand in detail how the system works in order to use it effectively. During installation of your Promethion system, all of the sensor modules and instruments were correctly addressed (see the chapter *Setting up the system*, below, for more details). Of course, you can change the address of any component or module of the system. You usually won't need or want to, but knowing that you can, for whatever reason, is reassuring.

When it comes to metabolic measurement, Promethion systems are available with two fundamentally different options. *These metabolic measurement options have no effect on the operation of the sensors associated with each cage.*

The first metabolic measurement option is the more common and economical. It splits one or two analyzers (typically) between several cages, with each analyzer taking brief "metabolic snapshots" of the animal in each cage at intervals of a few minutes. This is referred to as a "multiplexed" system.

The second, parallel option is less common, and uses one analyzer per cage, giving you continuous or nearly continuous metabolic data.

Either method requires that background or "baseline" air be measured for optimum accuracy. This is because metabolic measurements depend on knowing the effect the animal has on air pulled from its cage, and that effect is measured relative to background air around the cage, which therefore must also be accurately measured (Lighton, 2008).

## *Multiplexed (analyzer chains shared between subjects)*

Typically, multiplexed Promethion systems use one gas analyzer chain (a GA-3) and one eight-channel pull-mode flow generator with a single sample output (a FR-8) per eight cages. We recommend taking a baseline reading every four or eight cage readings, so a typical sequence will be:

```
baseline – cage 1 – cage 2 – cage 3 – cage 4 – baseline – cage 5 – cage 6 – cage 7 – cage 8
```

or

```
baseline – cage 1 – cage 2 – cage 3 – cage 4 – cage 5 – cage 6 – cage 7 – cage 8
```

which then repeats itself over and over again. The shorter the time spent sampling from each source, referred to as the "dwell time", the more rapidly the "metabolic snapshots" can be taken. We recommend dwell times of 30 to 120 seconds. 15 seconds, or even ~12 seconds, can be used with care and with consultation with Sable Systems. Using the recommended minimum of 30 seconds, and taking a baseline every 4 readings, a complete scan of all cages can be completed every 5 minutes. This is comparable to the time-constant of Promethion mouse cages (about 4.5 minutes). Using 15 seconds, and taking a baseline every 8 readings, a complete scan of all cages can be completed every 2.25 minutes. Because every group of 8 cages utilizes its own gas analyzer chain, this blazing-fast figure is independent of the number of cages in the system.

The software package used for data acquisition in multiplexed systems is currently MetaScreen; however, it is planned to phase out the use of MetaScreen in this application and replace it with SableScreen, which is basically an improved and more versatile version of MetaScreen. However, we anticipate that MetaScreen will continue to be used until at least the end of 2014.

During analysis, the individual "metabolic snapshots" from each cage are stitched together to create a pseudo-continuous record of the animal's metabolic rate with a 1-second interpolated temporal resolution. All sensor

data, which are natively acquired ever second, are perfectly synchronized with this interpolated metabolic record.

Here are some more details on each of the components of a Promethion multiplexed metabolic measurement system.

## The FR-8 Flow Generator

The FR-8 flow generator pulls a constant flow of air from up to eight cages at once, plus a baseline flow. It actually contains ten flow control systems, one for each cage plus one for baseline flow and one for generating the flow that is selected to be sent to the GA-3 gas analyzer (below). The flow rate is equal for all cages, and can be set by the user. It is typically 2000 mL/minute for mice or 3000 mL/minute for rats.

A traditionally designed multiple-channel pull-flow system would require one separate pump for each pulled air stream. That adds up to a lot of power, heat dissipation, wear and tear, and maintenance. This is one reason why multiplexed pull-mode respirometry systems are rare, in spite of their many advantages.

However, *only one air stream at a time* is actually sent to the gas analyzer in a multiplexed system. Based on this insight, we replaced all but one pump with a single vacuum pump into which all *non-selected* gas streams are pulled at a precisely controlled flow rate (see above). The non-selected air streams are mixed together, but this does not matter, because they are not analyzed. Only the air stream selected within the FR-8 by valves operating under the control of the Promethion software is pulled through the single sample pump, and its flow rate is monitored by an accurate, NIST-traceable primary mass flow meter. Because the single sample pump cycles through all of the flow streams, and pulls its highly accurate flow through the flow meters that normally allow the flow for that air stream into the vacuum to be measured and controlled, the calibration of those secondary flow meters can be monitored by the primary flow meter and corrected if necessary, making the flow system extremely accurate. This ingenious system combines high accuracy and self-calibration with low maintenance and minimal power consumption.

## The GA-3 Gas Analyzer

Note: Only Promethion systems shipped before 10/2013 have one or more GA3 analyzers. Functionally, each of these analyzers has a single analyzer blade. In more recent systems, they have been replaced by the GA3mX (see below). Functionally the analyzers are equivalent, but the new analyzers differ slightly in plumbing and calibration.

Each GA-3 gas analyzer samples from the air flow selected by one FR-8 flow generator. The air stream is directed through a manifold within the analyzer, from which a flow controller pulls out an aliquot of the main flow, typically at 250 – 400 mL/minute (settable by the user; this flow rate is not critical and does not affect metabolic calculations). This aliquot is then pushed through a water vapor analyzer, a CO2 analyzer, and an O2 analyzer. All of these devices are housed on a single “analyzer blade” within the GA-3’s cabinet. This makes the GA-3 very easy to service. The interior of the GA-3 is regulated at a constant 37 °C, which provides a stable environment for the analyzers and prevents condensation. The GA-3 also contains an accurate, high resolution barometric pressure analyzer. Under command via its CAN bus interface, the GA-3 can switch a chemical scrubber column

into the airstream. This scrubber column removes water vapor (and, optionally, CO2) from the airstream passing through it, and is used to calibrate the water vapor analyzer using the technique of O2 dilution (Lighton, 2008), and (optionally) to zero the CO2 analyzer (see the chapter on *Calibration*). This is the only time that the sample air stream is dried; the rest of the time, water vapor dilution of O2 and CO2 is compensated for mathematically, using data from the GA-3's water vapor and barometric pressure analyzers. Promethion is the only system capable of this feat.

The GA-3 is highly integrated, and takes up far less space than individual gas analyzers would. In addition, its integrated nature makes the lengths of plumbing runs linking its various analyzers as short as possible, minimizing lag effects that can complicate data analysis.

As this picture shows, this technique makes a metabolic phenotyping system far more compact. To the left, you see a circa 2007 eight channel Sable Systems metabolic phenotyping system. To the right, an eight channel Promethion system with superior performance and much, much easier setup. Apart from the scrubber column chemicals, the only predictably expendable part of the GA-3 is its O2 sensor, which is a fuel cell type, and should be replaced approximately every 18 – 24 months. This is normally done by Sable Systems as part of an annual scheduled site visit, but can also be easily done on-site by any technician.

## The GA-3mX Gas Analyzer

The GA-3mX gas analyzer samples from the air flow selected by multiple FR-8 flow generators. Each input air stream from an FR-8 is directed through a manifold, from which flow controllers pulls an aliquot or subsample of each flow, typically 250 – 400 mL/minute (settable by the user). These aliquots are then pushed through a water vapor analyzer, a CO2 analyzer, and an O2 analyzer – one for each FR-8's aliquot. The GA3mX can contain from one (GA3m1) to four (GA3m4) separate, interior analyzer "blades", each of which contains the above analyzers plus a subsampling system. This makes the GA-3mX very easy to service and to upgrade, for example from 8 to 16 cages. A single GA3mX can service up to 32 multiplexed cages – 8 per blade. Relative to legacy metabolic phenotyping systems, you can imagine the space savings that the system offers.

The interior of the GA-3mX is regulated at a constant 37 °C, which provides a stable environment for the analyzers and prevents condensation. The GA-3mX also contains an accurate, high resolution barometric pressure analyzer. Under command via its CAN bus interface, the GA-3mX can switch a chemical scrubber column into the airstream of any blade. This scrubber column removes water vapor (and, optionally, CO2) from the airstream passing through it, and is used to calibrate the selected blade's water vapor analyzer using the technique of O2 dilution (Lighton, 2008), and (optionally) to zero the blade's CO2 analyzer (see the chapter on *Calibration*). This is the only time that the sample air stream is dried; the rest of the time, water vapor dilution of O2 and CO2 is compensated for mathematically, using data from the GA-3mX's water vapor and barometric pressure analyzers. Promethion is the only system capable of this feat.

The GA-3mX is highly integrated, and takes up far less space than individual gas analyzers would. In addition, its integrated nature makes the lengths of plumbing runs linking its various sensors as short as possible, minimizing lag effects that can complicate data analysis.

Apart from the scrubber column chemicals, the only predictably expendable parts of the GA-3mX are its O2 sensors, which are a fuel cell type, and should be replaced approximately every 18 – 24 months. This is normally done by Sable Systems as part of a scheduled site visit, but can also be easily done on-site by any technician.

## *Parallel a.k.a. "Continuous" (one analyzer chain per subject)*

Parallel Promethion systems use one four-channel pull flow generator plus baseline, giving five independent flow paths (FR-4b), and a four-channel gas analyzer (GA3m4), per group of four cages. Metabolic recordings are continuous for all animals, with only an occasional, brief baseline measurement that can be interpolated across during analysis. Recording metabolic data in this way allows the "instantaneous transform" (Bartholomew et al., 1981) to be used, which radically improves the time resolution and thus the detailed structure of the metabolic data relative to multiplexed recordings. All sensor data are, of course, perfectly synchronized with this detailed record.

The software package used for data acquisition in parallel systems is SableScreen.

Here are some more details on each of the components of a Promethion parallel metabolic measurement system.

### The GA-3m4 Gas Analyzer

The GA-3m4 actually contains four complete, integrated gas analysis chains, also called "analyzer blades", within a single case the same size as the GA-3 (above). Each analyzer blade comprises a flow controller, a water vapor analyzer, a CO2 analyzer, and an O2 analyzer. The interior of the GA-3m4 is regulated at a constant 37 °C, and also contains an accurate, high resolution barometric pressure analyzer. The GA-3m4 has five air stream inputs; one for each channel of sample flow from a cage (inputs 7-10), and one for baseline air flow (input 6). Each of the GA-3m4's four analyzer blades can sample either from the appropriate channel of air flow from its associated cage, or from the baseline flow, under command via its CAN bus interface. The GA-3m4 can also, on command, switch a chemical scrubber column into the baseline airstream. This scrubber column removes water vapor (and, optionally, CO2) from the airstream passing through it, and is used to calibrate the water vapor analyzer of a given blade using the technique of O2 dilution (Lighton, 2008), and (optionally) to zero the CO2 analyzer of a given blade (see the chapter on *Calibration*). This is the only time that the sample air stream is dried; the rest of the time, water vapor dilution of O2 and CO2 is compensated for mathematically, using data from the GA-3m4's water vapor and barometric pressure analyzers.

The GA-3m4 is extremely highly integrated, and takes up a tiny fraction of the space that individual gas analyzers would. In addition, its integrated nature makes the lengths of plumbing runs linking its various analyzers as short as possible, minimizing lag effects that can complicate data analysis.

Apart from the scrubber column chemicals, the only predictably expendable part of the GA-3m4 is the O2 sensor on each analyzer blade, which is a fuel cell type, and should be replaced approximately every 18 – 24 months. This is normally done by Sable Systems as part of a scheduled site visit, but can also be easily done on-site by any technician.

## The FR-4b Pull Flow Generator

The FR-4b pull flow generator pulls four isolated, independent air streams at a controlled flow rate from four cages, plus a fifth gas stream that acts as the baseline and is sourced from the immediate environment of the cages. It therefore contains five independent flow generators that operate in parallel. Each flow generator is precisely calibrated at the factory against a highly accurate, NIST-traceable master reference flow meter. Each FR-4b operates in conjunction with one GA-3m4 analyzer.

## The FR-10+ Push Flow Generator

The FR-10+ pull flow generator pushes eight air streams at a controlled flow rate into eight sealed cages or other sealed respirometry chambers, plus two air streams that acts as baselines. It is primarily intended for use with third-party, push-mode "chambers" such as treadmills. Each push flow generator within the FR-10 is precisely calibrated at the factory against a highly accurate, NIST-traceable master reference flow meter. Air leaving each cage is plumbed to a GA-3m4 analyzer. Each FR-10 is intended for use with up to *two* GA-3m4 analyzers.

## "Background Baselining": The GA-3m5 Gas Analyzer

A refinement of the parallel approach, on which Sable Systems holds a pending patent, actually eliminates the baselines from the data altogether. It does so by swapping multiple analyzer chains in and out of the gas stream from each cage, allowing each analyzer to be calibrated and to measure baselines while another analyzer measures the air from the cage to which it is normally assigned. The data from the different analyzers are then seamlessly joined during data analysis, yielding a unique, completely continuous metabolic data record for each animal. The analyzer that accomplishes this task (in conjunction with Promethion software) is the GA-3m5, which contains five independent analyzer blades. Four of the blades are switchable between the air streams from cages 1 through 4, and the baseline air stream. The fifth blade can sample any one of the five air streams entering the GA-3m5 – the air streams from cages 1 through 4, or the baseline.

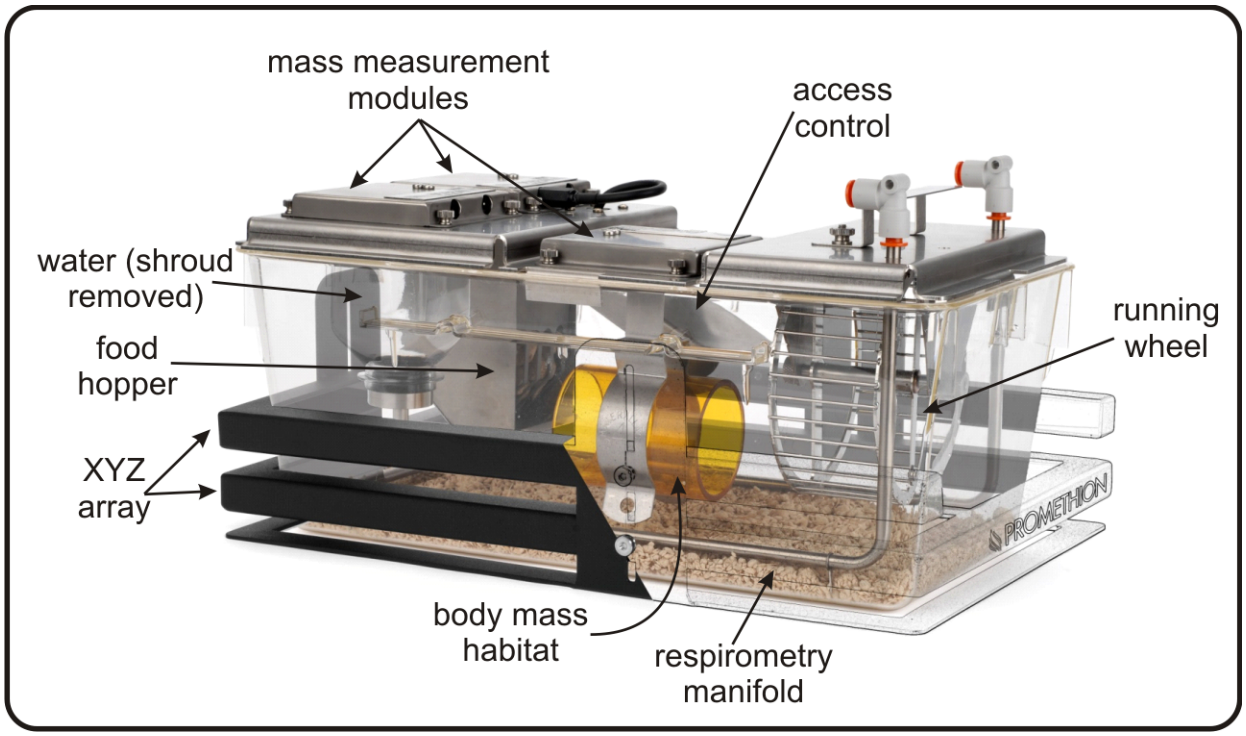
To understand how this works, imagine that blade 1 is sampling the air from cage 1. Now we are going to "background baseline" cage 1. We do this by switching blade 5 to sample the air stream from cage 1. It does so without interfering with blade 1's operation. Thus, both blades 1 and 5 are sampling the air from cage 1. Next, we switch blade 1 to baseline. This allows blade 1 to be calibrated, to measure background gas concentrations, and for any drift it may exhibit to be corrected. In the meantime, blade 5 continues to monitor the air stream normally assigned to blade 1. Blade 1 is then switched back to measuring the air from cage 1 again. Blade 5 continues to monitor that same air stream. Finally, blade 5 is switched to the baseline air stream so that it, too, can be calibrated and drift-corrected. During analysis, the data from the blades are "stitched" seamlessly to create a completely continuous, calibrated, drift-free set of data for each cage that allow the monitoring of second-by-second metabolic data with unprecedented accuracy and fidelity. This unique approach, which has

been validated in the most demanding respirometry application known – room calorimetry (Melanson et al., 2010) – is protected by a pending patent.

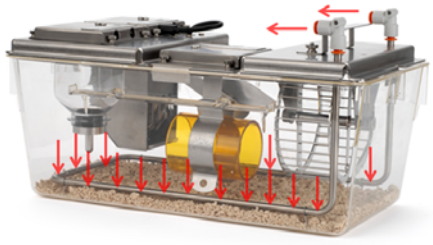
# LOCATION AND SETUP: CAGES

## *Background*

The animal cage is the fundamental unit of the Promethion system.




The above picture shows the mouse cage of a typical Promethion metabolic phenotyping system. (Rat systems are very similar, except that everything is scaled up in size to accommodate the larger animals.) As you can see, the typical cage uses three mass measurement modules which are typically configured to monitor one food intake, one water intake, and body mass. They can also be configured to monitor two food intakes and one water Intake. Optionally, access to one food container can be controlled. The position of the cage's inhabitant is monitored by the XYZ array. The animal is also free to run on the optional running wheel. Finally, a continuous flow of air is pulled from the respirometry manifold at the base of the cage. Micro-perforations along the length of that manifold ensure an even uptake of air from within the cage, so that all of the air breathed out by the animal is pulled from the cage for analysis.



As you can see, the cage is quite different from the typical small, un-enriched cage used by most metabolic phenotyping systems. As a true "home cage", it offers a spacious and enriched environment designed to minimize stress and acclimation time. Bedding may be used if desired. The bedding should be granular rather than fluffy so that the animal cannot pile it up and interfere with position detection.

Many researchers use the cage system for behavioral quantification and do not require metabolic measurement. In that case, the manifold will not be necessary. Alos a perforated lid is provided to ensure an adequate air exchange in the absence of pulled air flow.

Optionally, animals can be habituated to the cages by using a spare cage tub (supplied with the system). A perforated lid is used and hole covers with brackets take the placeof the mass monitoring devices. Food, water and body weight accessories are mounted on the hole covers.

The installation and calibration of the mass measurement modules and, and the installation and setup of the XYZ arrays are described in later chapters (see contents).

## *Housing the Cages*

The environment surrounding the cage is not critical. Ideally, it should be isolated from frequent human traffic in order to reduce stress on the experimental animals. Many researchers use wire racks to house the cages.

The ambient temperature does not affect the cage directly, but it can and does affect the cages' inhabitants. Typical animal care facilities are regulated between 21 and 23°C. This is comfortable for humans, but will cold-stress mice and rats, which have a far larger surface area to volume ratio than humans. Ideally, measurements should be conducted in the animals thermal neutral zone (TNZ)- for mice and rats around 28°C ambient temperature. Either the room temperature can be increased or the cages can be housed in a controlled-temperature cabinet.

As a minimum, experimental temperature should be reported as part of the experimental protocol. If more fine-grained resolution is required, Promethion systems can be augmented by one or more environmental monitors at the cages, which will continuously record temperature and relative humidity during the experiment.

## *Wiring the Cages*

Because all data transfer within the Promethion system is entirely digital, overall wiring complexity is minimized.

Wire each cage's cage controller module (CC-1) to each cage sensor (MM-1s, BXYZ) using a TRRS cables. Daisy-chain the cage controllers and connect them to the gas analyzers and flow controllers, if present using the CAT5 network cables. If you are using a running wheel, insert the reed switch in the hole adjacent to the wheel and plug its cable into the cage's CC-1. If you are using access control doors, connect each access control to the adjacent MM-1 using the short 2.5 mm jack attached to the access control. If you are using a wheel controller (wheel stop), attach the wheel controller to the CC-1 using a TRRS cable and connect one of the MM-1 outputs to the second slot on the wheel controller.

Each cage contains a digital network of its own. The private network of each cage is connected to a single device called a **cage controller** All of the sensors associated with a cage derive both their power and their data transfer from their associated cage controller. Each cage controller is connected to a power source, and to a data transfer cable that daisy-chains between all instruments and cage controllers within the system. Within each cage network, the individual sensors are connected to the cage controller via short connectors with four-way

plugs on each end. These are referred to as TRRS cables. They can be connected to the cage controller "DATA" sockets in any order. In the event that more sensors are connected to the cage, sensors can be daisychained without ill effect. Each sensor type broadcasts its identity in each data string. For sensors of the same type, such as mass measurement modules, the system distinguishes between different modules by their addresses.

All of the cage sensors were correctly addressed during system set up. You will not need to change them. For reference, however, here are the default sensors and their addresses.

| **Sensor Function** | **Module Type** | **Address** |
|---|---|---|
| Food Uptake Monitor A | MM-1 | 1 |
| Water Uptake Monitor | MM-1 | 2 |
| Body Mass Monitor *or* Food Uptake Monitor B | MM-1 | 3 |
| X array | BXY | 1 |
| Y array | BXY | 2 |
| Z array | BZ | 3 |

Each cage controller is always associated with a given cage, for example, cage 1. Each cage controller is addressed according to the number of the cage to which it is attached. Cage controller addresses can be changed, but in normal use this will not be necessary and could cause much conflict and confusion if it is done incorrectly. Please contact Sable systems if for any reason you need to change the addresses of your cage controllers.

The cage controllers are connected to the main Promethion network via daisy-chained Cat5 network cables. Thanks to this entirely digital data transfer protocol, adding additional sensors or cages to the system is very straightforward and incurs no penalty in the form of additional aggregators or interfaces.

If your system uses food access control doors, the access control door for each cage is mounted in front of food hopper A (address 1). The short cable attached to the access control door mount attaches to the smaller of the 2 sockets on food hopper A's MM-1.

If your system uses a running wheel, each running wheel contains a magnet that closes a contact on a reed switch with each revolution of the wheel. To install the reed switch, loosen the wing nuts that hold the running wheel in place. Gently insert the glass tube containing the reed switch into the small hole adjacent to the running wheel near the outside edge of the cage. Insert it until about 2 to 3 cm of the tube is above the cage top. Plug the cable attached to the reed switch into the CC-1 cage controller in the socket labelled "Wheel". Tighten the wing nuts. Please check for the correct operation of the reed switch with your software; follow the on-screen prompts and adjust the height of the reed switch if necessary while turning the wheel by hand. You might find it useful to attach a small piece of tape to the reed switch housing to denote the correct insertion depth.

The picture shows a typical 16 cage metabolic phenotyping system, of the multiplexed type. The 16 cages are divided into two groups of eight cages, each housed in its own temperature control cabinet. To the right of the temperature control cabinets, the flow generators and gas analyzers are visible within a four post instrument rack with a glass door. The blue air filters are visible at the top of each cage, as are the data cables, which also happen to be blue.

## *Plumbing the Cages*

For multiplexed systems, each cage is plumbed to an FR-8 pull flow controller via an in-line filter. Each FR-8 pull flow controller handles 8 cages plus a baseline. The baseline samples the air adjacent to the cages. See plumbing diagram 1. Each FR-8 pull flow controller sends one selected air stream to a GA3 or GA3mX gas analyzer chain. See plumbing diagrams 2 and 3.

For continuous systems, each cage is plumbed to an FR-4b flow controller via an in-line filter. Each FR-4b flow controller handles 4 cages plus a baseline. Each FR-4B flow controller sends five air streams to a GA3m4 gas analyzer.

Because the cages utilize pull flow respirometry, they are extremely easy to plumb. For each cage, the respirometry manifold is attached to a filter adapter (supplied) from which the cage air is pulled. Air is then pulled from the filter into the flow generators. The push-to-connect fittings on each cages respirometry manifold attach by a T adapter to the in-line air filter. Be sure that the arrow on the in-line air filter points away from the cage. A single length of tubing leads from the filter to the flow generator (FR-8).

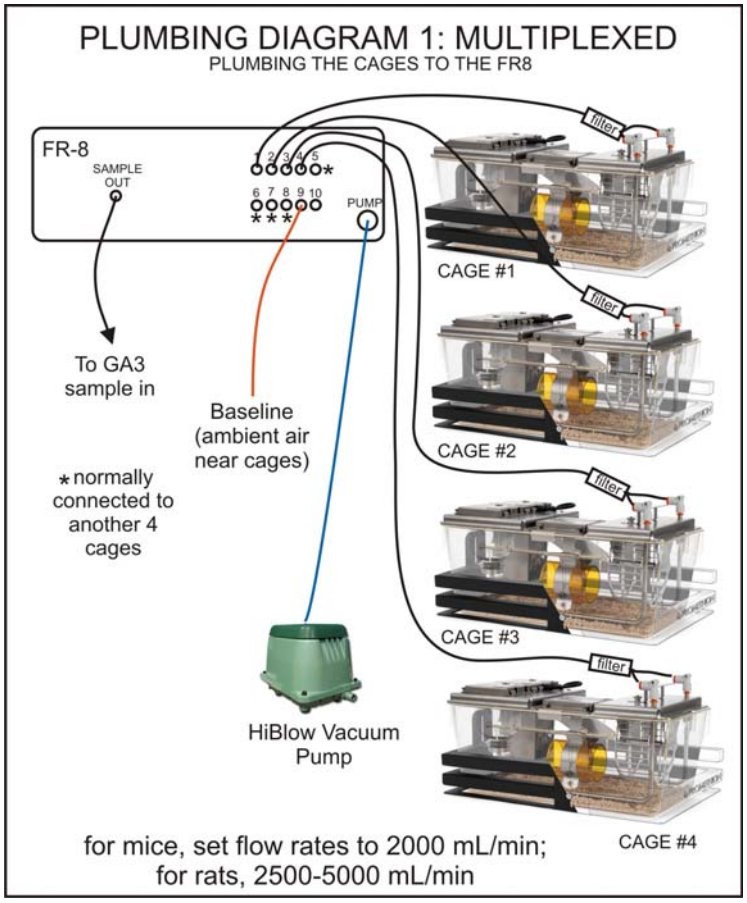
# LOCATION AND SETUP: ANALYZERS AND FLOW GENERATORS (MULTIPLEXED)

## *Background*

The analyzers and flow generators should all be housed in an instrument rack. The type of rack is not critical. For large systems (16 or more cages), a floor-mounting rack is suggested. For smaller systems, or where space is limited,, a desktop rack will work well.

A multiplexed system consists of one FR-8 pull flow generator for each 8 cages, plus one or more GA3 gas analyzers. In systems shipped prior to 2014, each FR-8 pull flow generator is attached to its own GA3 gas analyzer. In more recent systems, all FR-8 pull flow generators in the system are attached to one GA3mX gas analyzer that contains multiple gas analysis blades.

## *Plumbing the FR-8 Flow Generators to the Cages*




Plumbing diagram 1 shows how each cage is plumbed to an FR-8 pull flow generator. Only 4 cages are shown. This plumbing arrangement is identical in all systems. The outlet tubing is connected to the cage filter with a push-to-connect fitting.. Be sure that the arrow on the in-line air filter points away from the cage and toward the length of outlet tubing.

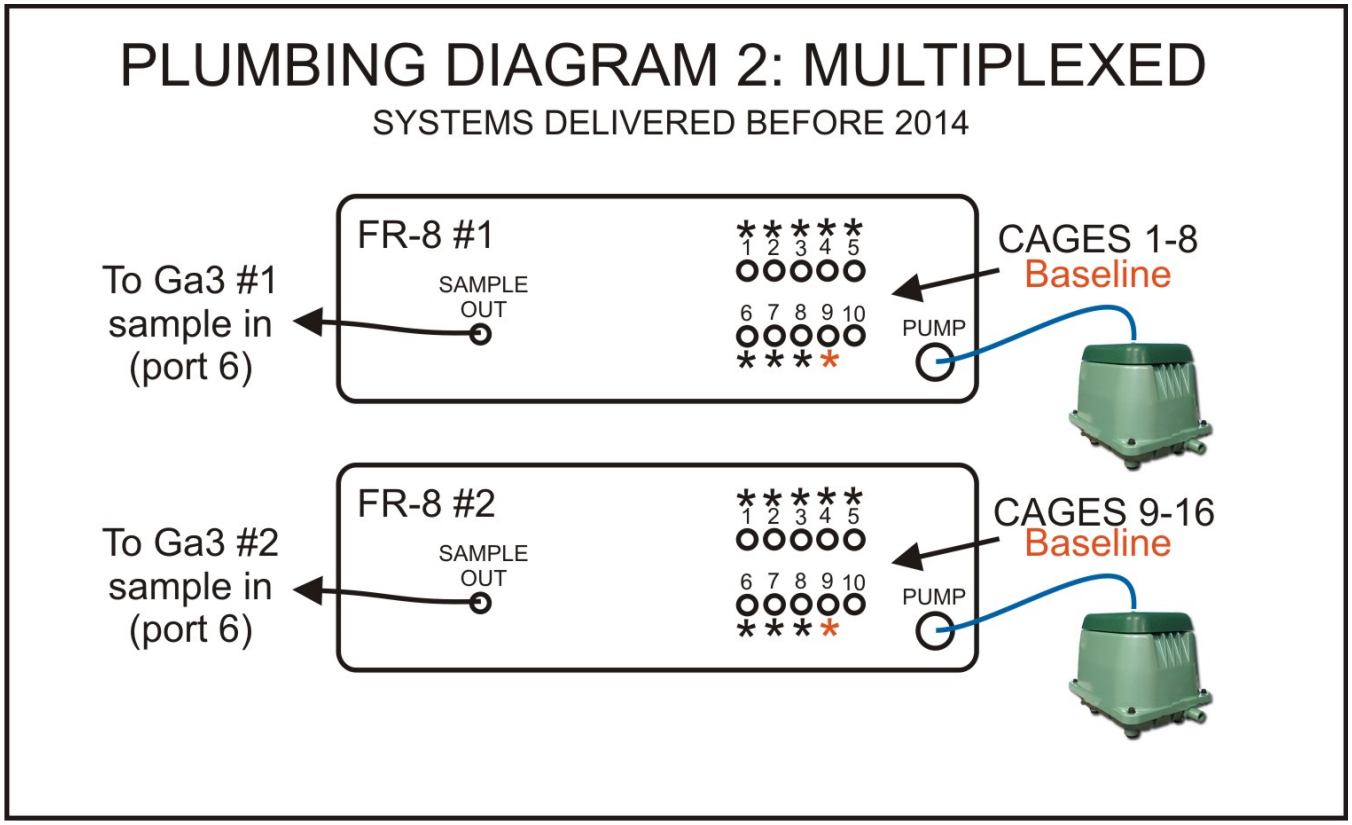
It is important that the tubing provided with the system (Bev-A-Line IV) should be used. This is because the majority of tubing in common use is not suitable for respirometry and may introduce errors.

The length of the tubing is not critical, but all cages should be attached to the flow generators with roughly equivalent lengths of tubing. If a tubing run of greater than about 3-4 meters is required, it will be advantageous to step up to a slightly larger tubing diameter, especially in the case of rat systems, which use a relatively high flow rate. Please contact Sable systems for detailed instructions if this is the case.

## *Plumbing the FR-8 Flow Generators to the GA3 Gas Analyzers*

### Pre-2014 Gas Analyzers

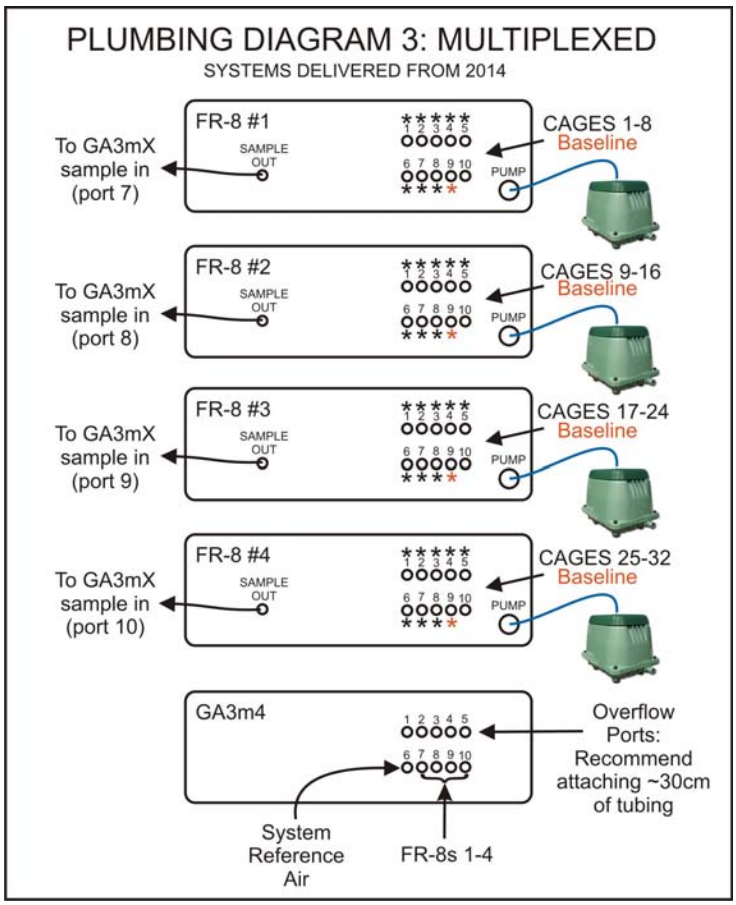
In multiplexed Promethion systems delivered prior to 2014, each FR-8 pull flow generator is plumbed to its own GA-3 gas analyzer. Please note that the system knows which FR-8 goes with which GA-3. These were correctly addressed during system set up, and should not be changed. Each GA-3 is addressed with an odd number starting with 1, while the FR-8 associated with each GA-3 is addressed with a number 1 greater than the address of its associated GA-3.




In the above diagram, 2 vacuum pumps are shown, one for each FR-8. Depending on the system configuration, two FR-8 may share a single vacuum pump.

### 2014 and Later Gas Analyzers

For systems delivered after late 2013, the plumbing is slightly different. For these systems, one FR-8 is present for every eight cages, exactly as before. However, the FR-8s are all plumbed to a single GA-3mX, where X is the number of eight cage groups in the system (1 for 8 cages, 2 for 16, and so on, up to a GA3m4 for 32 cages). This arrangement makes it easier to upgrade the system, is more space efficient, and ensures that all of the analyzer blades share a common environment to facilitate optimal inter-blade matching. The plumbing diagram is shown below.




Plumbing diagram 3 is a simplified representation of the new plumbing. The plumbing to the individual cages (black asterisks) and to the baseline air in the vicinity of the cages (orange asterisks) is not shown. See plumbing diagram 1 and the previous chapter for details. The GA-3m4 analyzer is shown at the bottom of plumbing diagram 3.

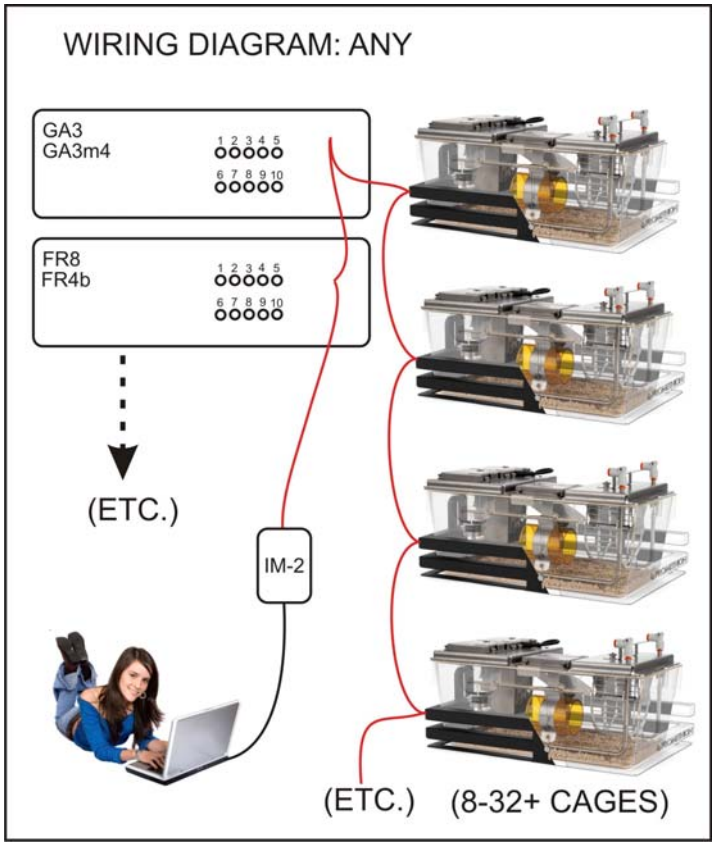
In plumbing diagram 3, 1 vacuum pump for each FR-8 is shown. Your system might use a single vacuum pump for two FR-8's.

## System Reference Air

There is a further important change in systems supplied from 2014 onwards. Notice that each FR-8 samples baseline from the vicinity of the cages to which it is attached. Those groups of cages may be in different locations, for example in different temperature cabinets or other environments that may differ in background gas composition. In order to ensure that all of the gas analyzer blades within the GA-3m4 are referenced against precisely the same gas concentrations in all cases, a separate system reference air source is used. This air source is only utilized during the initial calibration phase of each recording. For best results, care should be taken to ensure that the system reference air source is as stable as possible. A small air pump is supplied with the GA-3m4, and should be situated in an area well away from frequent traffic, such as behind the instrument rack. This pump is installed during system set up. A small buffer vessel is recommended for the pump input.The flow rate supplied to the GA3m4 is not critical, but should exceed 1.5 L per minute.

As an additional advantage, a separate system reference air source makes the GA-3m4 fully compatible with automated calibration of H2O, CO2, and O2 zero and CO2 span. (Note that H2O and CO2 zeros are automatically determined at the start of each recording, using the small scrubber column attached to the front of the GA3m4).

## *Wiring the FR-8 Flow Generators and the GA-3 Gas Analyzers*




Because the system utilizes advanced digital data transfer techniques, wiring the system is straightforward. Start at the computer and connect the system interface module (IM-2) using the supplied USB cable. Plug a CAT5 cable into the CAT5 cable socket on the IM-2. Make sure that the cable is long enough to reach the first instrument or cage to which it will be attached. If a cable run of > 5 m is required, we suggest that you run an active USB extension cable from the computer, and position the IM-2 close to the instrumentation and cages. In this way, a computer-to-system distance of some tens of meters can be achieved. Active USB extension cables include circuitry that extends the effective range of the USB protocol; do not use a simple USB extension cable without active circuitry in this long-range application. Contact Sable Systems if you need assistance in selecting an Active USB extension cable.

Continue to daisy-chain the entire system using CAT5 cables. The order of cages and instruments on this data bus is not important. However, the last instrument or the last cage in the chain will have one open CAT5 cable socket. **It is important to insert the provided network terminator plug into that socket.**

To extend or expand the system later, simply remove the network terminator plug, plug in the new components to your system, and finally, remember to insert the network terminator plug into the last open socket on the data bus.

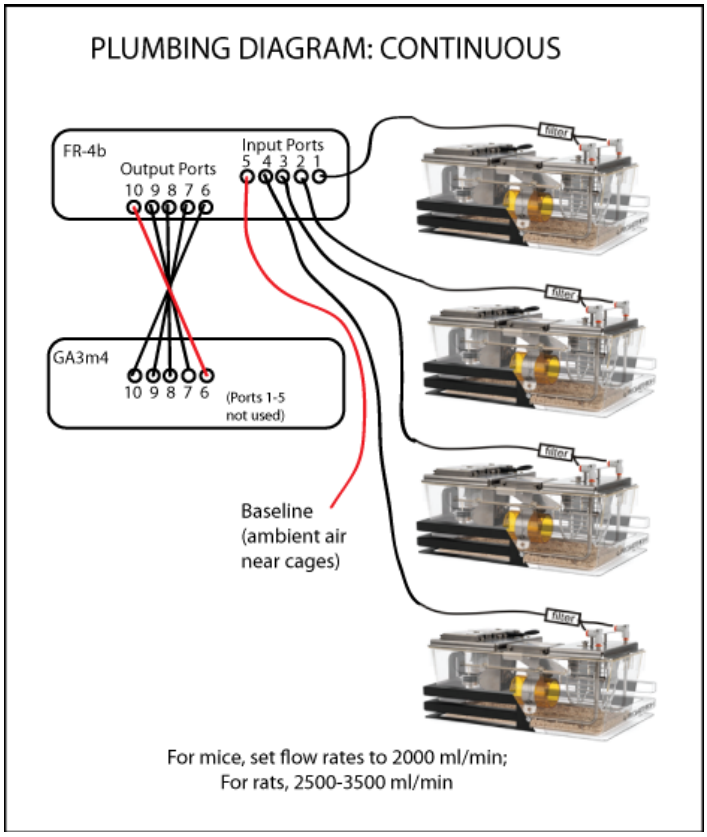
# LOCATION AND SETUP: ANALYZERS AND FLOW GENERATORS (CONTINUOUS)

## *Background*

The analyzers and flow generators should all be housed in an instrument rack. The type of rack is not critical. For large systems (8 or more cages), a floor-mounting rack is suggested. For smaller systems, or where space is at a premium, a desktop rack will work well.

A continuous system consists of one FR-4b pull flow generator, plus one GA3m4 gas analyzer, for each 4 cages.

## *Plumbing the FR-4b and GA3m4*



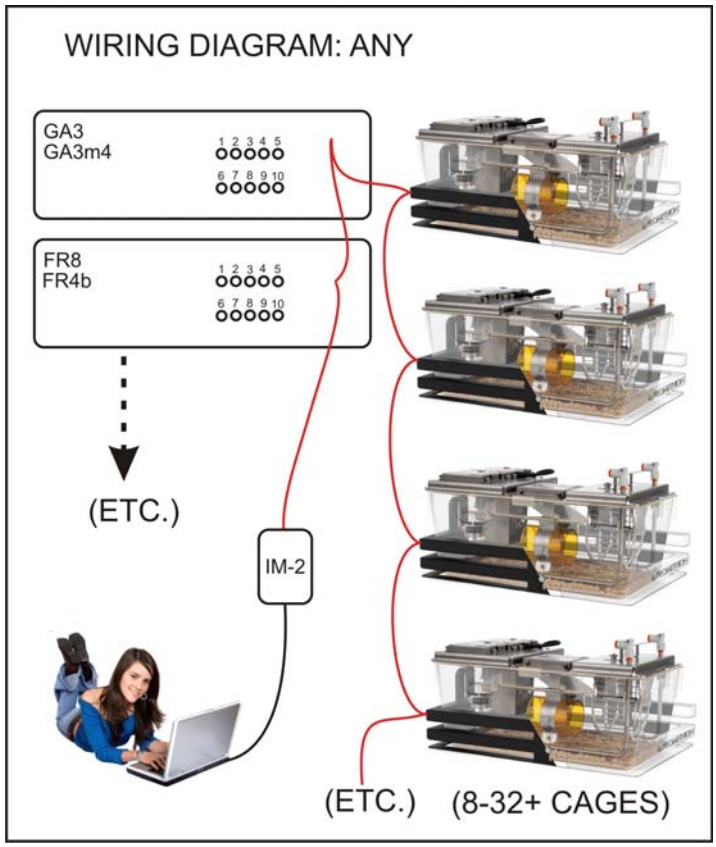

Plumbing diagram 4 shows a simplified view of the cage, flow generator, and gas analyzer plumbing of a 4-cage unit of a continuous metabolic phenotyping system. Each 4-cage unit is identical.

The outlet tubing is connected to the cage filter with a push-to-connect fitting.. Be sure that the arrow on the in-line air filter points away from the cage and toward the length of outlet tubing.

It is important that the tubing provided with the system (Bev-A-Line IV) should be used. This is because the majority of tubing in common use is not suitable for respirometry and may introduce errors.

The length of the tubing is not critical, but all cages should be attached to the flow generators with roughly equivalent lengths of tubing. If a tubing run of greater than about 3-4 m is required, it will be advantageous to step up to a slightly larger tubing diameter, especially in the case of rat systems, which use a relatively high flow rate. Please contact Sable systems for detailed instructions if this is the case.

## *Wiring the FR-4b Flow Generators and the GA-3m4 Gas Analyzers*




Because the system utilizes advanced digital data transfer techniques, wiring the system is straightforward. Start at the computer and connect the system interface module (IM-2) using the supplied USB cable. Plug a CAT5 cable into the CAT5 cable socket on the IM-2. Make sure that the cable is long enough to reach the first instrument or cage to which it will be attached. If a cable run of > 5 m is required, we suggest that you run an active USB extension cable from the computer, and position the IM-2 close to the instrumentation and cages. In this way, a computer-to-system distance of some tens of meters can be achieved. Active USB extension cables include circuitry that extends the effective range of the USB protocol; do not use a simple USB extension cable without active circuitry in this long-range application. Contact Sable Systems if you need assistance in selecting an Active USB extension cable.

Continue to daisy-chain the entire system using CAT5 cables. The order of cages and instruments on this data bus is not important. However, the last instrument or the last cage in the chain will have one open CAT5 cable socket. **It is important to insert the provided network terminator plug into that socket.**

To extend or expand the system later, simply remove the network terminator plug, plug in the new components to your system, and finally, remember to insert the network terminator plug into the last open socket on the data bus.

# CALIBRATION

## *Water Vapor Analyzer Calibration (automated)*

### Background

All Promethion gas analyzers contain a sensitive, stable water vapor pressure analyzer. The system uses the readings from the water vapor pressure analyzer, together with data from the analyzer's barometric pressure sensor, to perform water vapor dilution compensation for O2 and CO2 concentrations, plus flow rate correction for the presence of water vapor.

Calibrating water vapor analyzers is legendarily difficult. The zero point is relatively easy; a well-desiccated air stream is all that is required. However, spanning the analyzer requires a known water vapor pressure, and this is tricky to generate on demand. Your Promethion system uses a technique called O2 dilution (Lighton, 2008) to calibrate the span of its water vapor analyzer.

The principle is simple. First, baseline air is switched to flow through the analyzer chain. This air contains water vapor. The analyzer's desiccant column is then switched into the circuit, so that the air stream now contains no water vapor. Because the worldwide fractional concentration is extremely close to 0.2094 in dry air, the O2 analyzer can now be spanned to that value (after barometric pressure correction to the standard pressure of 101.325 kPa). This is a more accurate standard O2 concentration, even inside a typical building, than all but the most accurate, and most difficult and expensive to obtain, span gases (Tohjima et al., 2005). The water vapor pressure analyzer can also be zeroed at this point. Next, the scrubber column is switched back out of circuit. Water vapor now dilutes the O2 signal, and the measured O2 concentration falls. The water vapor pressure present in the air can now be calculated from the magnitude of this reduction in measured O2 concentration (Lighton, 2008). The water vapor pressure analyzer can then be automatically spanned to the calculated value.

Because CO2 can also be chemically scrubbed from the air stream at the same time, the Promethion system takes double advantage of this opportunity and also zeroes the CO2 analyzer during automated water vapor analyzer calibration. (This feature can be overridden, as can all automated calibration, but this is not recommended.)

The water vapor calibration system is critically dependent on the makeup of the chemical scrubber (calibration) column. Fortunately, the column is easy to make and lasts for a long time (many months) in typical use.

### Checking the state of the calibration column

Your scrubber (calibration) column should remove water vapor and CO2 from the air passing through it. You can check this from within MetaScreen as follows:

1. Start up MetaScreen. Navigate to the CALIBRATION tab.
2. Click CALIBRATE on MetaScreen. This switches the scrubber (calibration) column into the analyzer's flow path.

3. You should see the CO2 readings decline rapidly to approximately zero.

4. The water vapor readings should also drop to approximately zero, but much more slowly than the CO2.

5. When you are done, click STOP on MetaScreen. **Do this as soon as you are done. NEVER leave the scrubber column activated for longer than absolutely necessary.**

If either the CO2 or water vapor readings stabilize far from zero (more than ~±0.005% for CO2 or ~±0.03 kPa for water vapor pressure), then the chemical scrubbers in the column may need to be replaced. Before doing so, however, check that the N2 zero (below) is satisfactory. If the scrubber column shows higher numbers than the true-zero N2, then its chemicals definitely need to be replaced.

## Filling the Calibration Column

The column, which was supplied with the system and filled for you during system setup, is filled with a layer of desiccant, followed by a layer of CO2 absorbent, followed by another layer of desiccant. **IMPORTANT: Each layer *must* be separated by a fiber disc** or they will react; you do not want that to happen. Extra fiber discs, together with a spare column, are included with the system and can also be obtained directly from Sable Systems.

Again, we stress that a properly constructed calibration column will last for many months. Remember that it is usually only used for a few minutes at the start of each recording.

For the best possible results, we recommend using magnesium perchlorate as the desiccant. If you do, be sure to use a granular rather than powdery grade so that air can flow through it without much resistance. Magnesium perchlorate is a superb desiccant but does have the following disadvantages:

- It is toxic, caustic and a very strong oxidizer (in fact it is used for that purpose in solid rocket motors).
- It can cause serious skin burns if left in contact with the skin.
- It does not contain an indicator, and it is very difficult to determine its hydration state by inspection until it is exhausted, at which point it deliquesces and begins to attack its surroundings. There are, however, ways to check the column's state (see below).

For these reasons, many users may prefer to use Drierite, which contains an indicator and gives acceptable results. For optimum results the Drierite should be 20-40 mesh (Drierite stock number 24001 or 24005). At least the top layer of the column should be changed as soon as the top desiccant layer's indicator reaches more than ~30% down to the CO2 scrubber layer. Don't be tempted to regenerate it by heating; the temperature is critical or the Drierite will "die", plus the regenerated Drierite is powdery and loses most of its indicating capability.

To change the chemicals in the column:

1. If the Ascarite in the column is fresh (the whitening indicative of exhaustion is no more than ~30% down the Ascarite layer, and the column's CO2 absorption capability is good; see below), you need only change the top layer of desiccant. In that case, do so, decanting the top desiccant carefully and replacing

it (see the notes on desiccator precautions, below). Put a new fiber circle on the top and replace the column's top cap. Make sure that the column is neither packed to loosely (no visible voids) nor too tightly (the cap should fit on easily). If, however, you are replacing the entire column:

2. Empty the column. Bear in mind that if you are using magnesium perchloride as the desiccant, it is caustic, toxic, and a very strong oxidizer, and you should use standard precautions to avoid any skin contact or breathing any dust. Also, the CO2 absorbent, Ascarite, contains NaOH and should be handled with similar precautions. Dispose of the chemicals in accordance with local regulations.
3. Fit the bottom cap onto the column.
4. Place a new fiber washer over the cap.
5. Fill the column to about 25% with desiccant.
6. Place a new fiber washer over the bottom desiccant layer.
7. Fill the column to about 60% (total) with Ascarite.
8. Place a new fiber washer over the Ascarite layer.
9. Fill the rest of the column, leaving a bit of room for the final fiber washer, with desiccant. Tap the column on the benchtop to settle it before adding the fiber washer.
10. Place a new fiber washer over the top desiccant layer.
11. Make sure that the column is neither packed to loosely (voids should not be visible, or "channeling" can occur, leading to in complete scrubbing) nor too tightly (the cap should fit on easily; packing the column too tightly may obstruct air flow).
12. When you attach the column to the analyzer, make certain that the connections are screwed all the way in. A touch of vacuum or stopcock grease on the interior of the analyzer's scrubber column ports helps.

## *Gas Calibrations (Manual)*

Manual calibration is quick and easy with a bit of practice. Please have the following items on hand:

1. A tank of pure analytical nitrogen with a two-stage regulator.
2. A tank of CO2 span gas, typically 0.5% CO2, balance N2 or N2 + O2, also with a two-stage regulator. The CO2 span gas should be in the range of 0.25 – 1%, with 0.5% being ideal. The precision of the gas should be 2% or better. Remember that the precision with which you can measure RQ is dependent on the accuracy of your CO2 span gas.
3. Each regulator should be attached to a needle valve or any other style of adjustable valve that can regulate the flow rate of the gas.

4. The gas flow from each needle valve should be brought out to ¼" (6 mm) o.d. tubing long enough to reach the analyzer being calibrated.

The exact flow rate of the two gases is not very critical, but should not be less than approximately 500 mL/minute or more than approximately 2 L/minute. With some experience you can learn to estimate approximate flow rates without using a flow meter (if you direct the flow onto your lips, it should be easily felt, but nowhere near fast enough to be heard emerging from the tubing). If you are not experienced enough with regulating gas flows intuitively, then you will need a simple flowmeter such as a rotameter. The simplest and cheapest is a so-called in-line rotameter or "pocket flowmeter" such as this:

This type of flowmeter is available in several ranges. The most suitable is approximately 200 – 2000 mL/minute of air (or about 0.5 to 5 SCFH). An example is part number EW-32500-80 from Cole Parmer (coleparmer.com; also available from Amazon.com). Be sure that air (or any other common gas such as O2 or N2) is explicitly cited as the metered fluid. The meter needs to be held upright to obtain an accurate reading. You can also purchase a rotameter with a stand. Because the accuracy requirement for calibration gas flow is not stringent, there is no need to invest in a particularly accurate (and thus expensive) rotameter. A suitable device will cost from approximately $20 (in line) to less than $100 (stand-alone with built-in needle valve; for example, the Cole Parmer WU-68560-22, $58 in 2013). Be sure that the meter has an output port (some pocket flowmeters intended for monitoring flow from O2 generators do not have a usable output port and thus cannot be used in this application).

You will need tubing connectors. These are supplied by Sable Systems during setup, but you can also obtain them from most scientific or industrial supply houses. We particularly recommend Luer-Lok connectors, which use the familiar tapered connection used by syringes and needles. These connectors are available with barbed fittings for attaching to ¼" o.d., 1/8" i.d. (6mm o.d., 3 mm i.d.) tubing. They facilitate quick, easy and secure connections. If you are purchasing tubing (again, Sable Systems will usually supply suitable tubing during setup), you should obtain semi-rigid tubing with a hydrophobic fluoropolymer lining such as Bev-A-Line. Soft tubing such as Tygon or silicone is too gas-permeable to give good results, especially with regard to zeroing the water vapor analyzer.

## Multiplexed

First, check on the state of health of your scrubber (calibration) column; see above.

Now for the actual calibration. We recommend doing this on a weekly basis until you are familiar with your system. If you find that the system's drift is negligible you can extend the interval between calibrations.

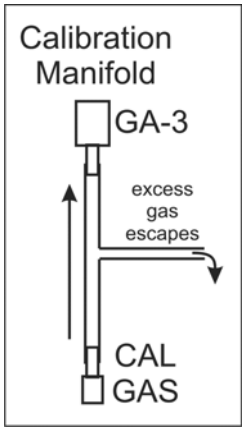
Because of design changes, the manual calibration procedure is slightly different for multiplexed Promethion systems of different ages. There are three alternatives; one for a small number of systems delivered before June of 2011, one for systems delivered prior to August 2013, and the other for later systems.

**For systems delivered prior to June 2011:**

1. Start up MetaScreen. Navigate to the CALIBRATION tab.
2. Disconnect the desiccator (calibration) column on the front of the GA3.
3. Start N2 flowing at about 500 - 1000 mL/minute.
4. Click CALIBRATE on MetaScreen. **Do not omit this step; it is essential before you proceed.**
5. Attach the N2 (zero CO2) stream to the UPPER port to which the column was attached. Use the threaded Clippard quick-connect fitting that we supplied during system setup. If this has been misplaced, you can obtain more from Sable Systems or from Clippard (clippard.com; part MQC-FS).
6. You should see the CO2 readings decline rapidly to approximately zero.
7. When the CO2 readings have stabilized and ceased significant downward drift (about a minute), click on ZERO CO2 and confirm your choice. **NEVER zero the CO2 analyzer on anything but known-zero CO2 gas!**
8. Optionally, you can also check on the zero readings of the water vapor analyzer and the O2 analyzer. If you do so, you may need to flow N2 for about 5 minutes (water vapor) or 20 minutes (O2) before a stable zero is obtained. Each analyzer has an associated zero button on-screen, as for CO2.
9. Stop the N2. Start the CO2 span gas flowing at about 500 - 1000 mL/minute.
10. Attach the CO2 span gas stream to the UPPER port to which the column was attached.
11. You should see the CO2 readings increase rapidly to a stable value.
12. Enter the actual value of your CO2 span gas in the text box provided on the screen.
13. When the CO2 readings have stabilized, click on SPAN CO2 and confirm your choice.
14. Turn off the CO2 span gas. **IMPORTANT: BE SURE TO DO THIS. Span gases are expensive!**
15. You may wish to re-check the zero with N2. It should be stable. If not (which would be unusual), repeat the above steps.
16. When you are done, click STOP on MetaScreen. **Do this as soon as you are done.**
17. Re-connect the scrubber column.
18. Check the scrubber column's action as described above. When the scrubber column is switched into circuit, CO2 and water vapor pressure readings should decline to values similar to those obtained with N2. **Remember to click STOP as soon as you are done. NEVER leave the scrubber column activated for longer than absolutely necessary.**

**For systems delivered after June 2011 and before August 2013,** the procedure is slightly different:

1. Start up MetaScreen. Navigate to the CALIBRATION tab.
2. Disconnect the desiccator (calibration) column on the front of the GA3.
3. Start N2 flowing at about 1000 mL/minute.
4. Click CALIBRATE on MetaScreen. **Do not omit this step; it is essential before you proceed.**
5. Attach the manifold that we supplied during setup to the UPPER port to which the column was attached. The manifold consists of a Luer-Lok connector to which the calibration gas stream is attached, followed by a T-connector. The T-connector is attached on the opposite end to a threaded Clippard quick-connect fitting. The middle branch of the T-connector is attached to a ~35 cm length of tubing to which nothing else is connected. We supplied this manifold during system setup. It is diagrammed here. We are using a manifold, which allows excess gas to escape, rather than pushing gas directly into the GA3 as in older systems, because the gas stream is now pumped into the GA3 from the top calibration column port. If the manifold has been misplaced, you can obtain another from Sable Systems. Alternatively, you can make your own from Bev-A-Line tubing (1/4" or 6 mm o.d.), a female Luer-Lok connector (McMaster-Carr part number 51525K213 or equivalent, mcmaster.com), a T-connector sized for your tubing (e.g. McMaster-Carr part number 51055K986 or equivalent, mcmaster.com), and a Clippard Quick-Connect (clippard.com; part MQC-FS).




6. Connect the N2 (zero CO2) stream to the manifold.
7. You should see the CO2 readings decline rapidly to approximately zero.
8. When the CO2 readings have stabilized and ceased significant downward drift (about a minute), click on ZERO CO2 and confirm your choice. **NEVER zero the CO2 analyzer on anything but known-zero CO2 gas!**
9. Optionally, you can also check on the zero readings of the water vapor analyzer and the O2 analyzer. If you do so, you may need to flow N2 for about 5 minutes (water vapor) or 20 minutes (O2) before a stable zero is obtained. Each analyzer has an associated zero button on-screen, as for CO2.
10. Stop the N2. Start the CO2 span gas flowing at about 1000 mL/minute.
11. Attach the CO2 span gas stream to the manifold.
12. You should see the CO2 readings increase rapidly to a stable value.
13. Enter the actual value of your CO2 span gas in the text box provided on the screen.
14. When the CO2 readings have stabilized, click on SPAN CO2 and confirm your choice.

15. Turn off the CO2 span gas. **IMPORTANT: BE SURE TO DO THIS. Span gases are expensive!**

16. You may wish to re-check the zero with N2. It should be stable. If not (which would be unusual), repeat the above steps.

17. When you are done, click STOP on MetaScreen. **Do this as soon as you are done.**

18. Re-connect the scrubber column.

19. Check the scrubber column's action as described above. When the scrubber column is switched into circuit, CO2 and water vapor pressure readings should decline to values similar to those obtained with N2. **Remember to click STOP as soon as you are done. NEVER leave the scrubber column activated for longer than absolutely necessary.**

**For systems delivered after August 2013,** the procedure is slightly different again:

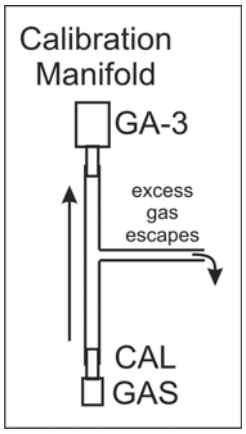
1. Start up MetaScreen. Navigate to the CALIBRATION tab.

2. Disconnect the desiccator (calibration) column on the front of the GA3.

3. Start N2 flowing at about 1000 mL/minute.

4. Click CALIBRATE on MetaScreen. **Do not omit this step; it is essential before you proceed.**

5. Attach the manifold that we supplied during setup to the UPPER port to which the column was attached. The manifold consists of a Luer-Lok connector to which the calibration gas stream is attached, followed by a T-connector. The T-connector is attached on the opposite end to a threaded Clippard quick-connect fitting. The middle branch of the T-connector is attached to a ~35 cm length of tubing to which nothing else is connected. We supplied this manifold during system setup. It is diagrammed here. We are using a manifold, which allows excess gas to escape, rather than pushing gas directly into the GA3 as in older systems, because the gas stream is now pumped into the GA3 from the top calibration column port. If the manifold has been misplaced, you can obtain another from Sable Systems. Alternatively, you can make your own from Bev-A-Line tubing (1/4" or 6 mm o.d.), a female Luer-Lok connector (McMaster-Carr part number 51525K213 or equivalent, mcmaster.com), a T-connector sized for your tubing (e.g. McMaster-Carr part number 51055K986 or equivalent, mcmaster.com), and a Clippard Quick-Connect (clippard.com; part MQC-FS).




6. Connect the N2 (zero CO2) stream to the manifold.

7. Select a blade to calibrate (1 through 4; your GA3mX might have fewer blades if it used in a multiplexing system. It should have one blade for every FR-8 connected to it).

8. You should see the CO2 readings of the selected blade decline rapidly to approximately zero.
9. When the CO2 readings have stabilized and ceased significant downward drift (about a minute), click on ZERO CO2 and confirm your choice. **NEVER zero the CO2 analyzer on anything but known-zero CO2 gas!**
10. Optionally, you can also check on the zero readings of the water vapor analyzer and the O2 analyzer. If you do so, you may need to flow N2 for about 5 minutes (water vapor) or 20 minutes (O2) before a stable zero is obtained. Each analyzer has an associated zero button on-screen, as for CO2.
11. Stop the N2. Start the CO2 span gas flowing at about 1000 mL/minute.
12. Attach the CO2 span gas stream to the manifold.
13. You should see the CO2 readings increase rapidly to a stable value.
14. Enter the actual value of your CO2 span gas in the text box provided on the screen.
15. When the CO2 readings have stabilized, click on SPAN CO2 and confirm your choice.
16. Turn off the CO2 span gas. **IMPORTANT: BE SURE TO DO THIS. Span gases are expensive!**
17. You may wish to re-check the zero with N2. It should be stable. If not (which would be unusual), repeat the above steps.
18. Move on to the next blade. Repeat the above steps.
19. When you are done, click STOP on MetaScreen. **Do this as soon as you are done.**
20. Re-connect the scrubber column.
21. Check the scrubber column's action as described above, with each blade. When the scrubber column is switched into circuit, CO2 and water vapor pressure readings on the selected blade should decline to values similar to those obtained with N2. **Remember to click STOP as soon as you are done. NEVER leave the scrubber column activated for longer than absolutely necessary.**

## Parallel

First, check on the state of health of your scrubber (calibration) column (see above).

Now for the actual calibration. We recommend doing this on a weekly basis until you are familiar with your system. If you find that the system's drift is negligible you can extend the interval between calibrations. You will calibrate each blade in your GA3m4 or GA3m5 individually.

1. Start up SableScreen. Navigate to the UTLILITES TAB and then to the CALIBRATION menu item.

2. Start N2 flowing at about 1000 mL/minute.

3. Attach the N2 (zero CO2) stream to the BASELINE port of the GA3m4 or GA3m5, which is port #1 on the back of the instrument..

4. In the CALIBRATION screen click on the BASELINE button next to the analyzer module you want to calibrate.

5. You should see the CO2 readings decline rapidly to approximately zero.

6. When the CO2 readings have stabilized and ceased significant downward drift, click on ZERO CO2 and confirm your choice. Sable Screen enforces an appropriate minimum wait time by disabling the zero buttons for the countdown indicated. **NEVER zero the CO2 analyzer on anything but known-zero CO2 gas!**

7. Optionally, you can also check on the zero readings of the water vapor analyzer and the O2 analyzer. If you do so, you may need to flow N2 for about 5 minutes (water vapor) or 20 minutes (O2) before a stable zero is obtained. Each analyzer has an associated zero button on-screen, as for CO2.

8. Stop the N2. Start the CO2 span gas flowing at about 1000 mL/minute.

9. Attach the CO2 (span CO2) stream to the BASELINE port of the GA3m4 or GA3m5, which is port #1 on the back of the instrument. You should see the CO2 readings increase rapidly to a stable value relatively close to the span concentration.

10. Enter the actual value of your CO2 span gas in the text box provided on the screen.

11. When the CO2 readings have stabilized, click on SPAN CO2 and confirm your choice.

12. Turn off the CO2 span gas. **IMPORTANT: BE SURE TO DO THIS. Span gases are expensive!**

13. Switch the just calibrated module back to sample flow (button SAMPLE), the live readings should freeze.

14. Move on to the next blade and repeat steps 2-13 as necessary

15. Check the scrubber column's action as described above. **Remember to switch back to sample flow as soon as you are done. NEVER leave the scrubber column activated for longer than absolutely necessary.**

    NOTE: It is also possible to zero and span multiple modules at the same time. Make sure to increase the flow of gases accordingly. E.g. for 4 modules a minimum flow of 2000 mL/min is required. A good strategy is to switch the modules to baseline flow sequentially and wait for the CO2 concentration to drop. Switching in another module should not stop the decrease of the concentrations in other modules. Otherwise, the flow of gas is too low for a reliable calibration and should be increased.

# BASIC DATA ANALYSIS 1: EXTRACTING THE DATA YOU NEED

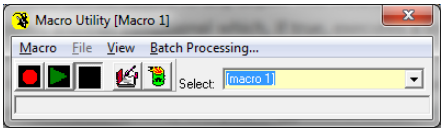
Your Promethion system records data in raw form, and for optimal accuracy and versatility the actual analysis of data in final form takes place during data analysis. To guide you and provide feedback during the acquisition of data, some data are calculated in real time during data acquisition, but these values will be less accurate than values calculated during data analysis. There are several reasons for this, but the most important is as follows: When values are calculated in real time, the future values of incoming raw data are unknown. However, knowing those values allows much more accurate compensation for analyzer drift, shifts in incurrent gas concentrations, and so on. In the absence of a means of predicting the future, it is therefore impossible to attain optimal accuracy with data analyzed in real time.. As a firm founded by scientists, Sable Systems appreciates your requirement for accuracy, versatility and transparency. Our extensive set of data analysis tools supports your scientific requirements, facilitating the transparency and replicability of your data and analysis methods.

## Running a Macro (A Step by Step Guide)

Data analysis is accomplished by running a sequence of analysis scripts called a Macro Collection.  Am ExpeData macro is a series of operations that can be performed in sequence without additional intervention from the operator. This section shows you how to run a typical macro collection (typically called a Universal Macro Collection), that is supplied with your system and may be updated periodically.

*BEFORE YOU START:* Make sure you have the latest release of ExpeData, or certain recently added features of the macro language may not work. Also be sure to run the macro on a computer with high speed and plenty of RAM (8 GB and more recommended, especially for larger files).

1. Load your raw data file, for example MyData.exp (FILE/OPEN).
2. Open the macro window (TOOLS/MACRO UTILITY).
3. Load the appropriate macro collection (in the macro window, MACRO/OPEN).
4. A macro collection consists of several individual, separate macros. By default, macro 1 will be active after you load a macro collection. You can see which macro is active from the SELECT drop-down box (macro 1 in the example). If you want to see the function of each macro, click VIEW. If you want to read the text for any macro, select it and then click on the notepad icon in the macro window.




5. Your macro collection is described in the VIEW window. The macro numbers in your macro collection may be different from those shown here. Typical instructions for running a macro follow.
6. For extracting only behavioral data (EthoScan), run a suitable macro on the raw data file. Macros that extract behavioral data require raw data.
7. Most other data extraction follows a two-step process. First, a pre-processing macro is run. Usually this is macro 1. When the preprocessing macro finishes, it saves the data in its transformed form. When you

need to extract data again, simply load the preprocessed file to save time. Second, run a macro on the preprocessed file to extract the data you require.

8. On a raw data file, start by running macro 1. Click the green sideways “PLAY” triangle.

9. Macro 1 will run. Depending on the size of your file and the speed of your computer, it may take several minutes to complete. When it is complete, a dialog box appears that announces that it has finished. It also saves your data file, now transformed with values for the VO2, VCO2 and RQ, etc., of each animal, as the root file name appended with “_m”, for example MyData_m.exp. *NEVER save your transformed data over your original raw data file!*

10. You are now able to extract data from your transformed file. Click VIEW in the macro window to see the macros that are available. Double-click on the description of the macro you want to use. Each macro in the collection extracts data in different formats and for different purposes. For most users, the most useful are macros 2 and 3 (for multiplexed data; extract summarized data for each cycle interval of the metabolic measurement system, i.e. every 2 – 20 minutes, depending on settings during data acquisition), or macro 5 (extract data summarized by circadian cycle; for either multiplexed or continuous data). Click whichever one you wish.

    a. Note that macros 4 and 5 can easily be edited to allow for different circadian cycle times. Alternative versions for different circadian cycle start times may be included in your macro collection. Please see the macro text for details.

11. The macro runs, and data are extracted into ExpeData’s spreadsheet. To open the spreadsheet (i.e. make it visible), in the main ExpeData window click on VIEW/SPREADSHEET. The spreadsheet can be open or closed when data are being written into it; it makes no difference. **Important note**: Once the spreadsheet has been ‘opened’ with data in it, you must save the data before you close the spreadsheet. If you attempt to close a spreadsheet with unsaved data in it, the program will prompt you to save the data before closing the spreadsheet.

12. Note that for very large data sets, it is also possible to run a macro that extracts data straight into Excel without passing through ExpeData’s built-in spreadsheet, which has row and column limitations (unlike Excel). See the VIEW window for the number of this macro, which saves a lot of time, but does require that Excel is installed on the computer. If Excel is not installed, the macro will complain, but the data will still be copied to the clipboard from which it can be pasted into other applications.

13. You can export data from the ExpeData spreadsheet in three different ways.

    a. Save the data as a .csv file (FILE/SAVE). Be sure to specify commas as the delimiter in the spreadsheet’s main window before you save. Give the file a .csv extension so that other programs will recognize it.

    b. Export the data directly to Excel. Click EDIT/EXPORT TO EXCEL. A new instance of Excel will be created and the data will be written to it. Note that date and time data will be “scrunched” into

hashmarks; to re-size all columns so that they are formatted correctly, click on the small diagonal arrow in the top left corner of the spreadsheet to select all the data, then position the cursor over the border of any two column headings (A, B, C etc.) and click it. All rows will now be correctly re-sized. If you want time in Excel to be formatted to the second, not just to the minute (the default), right click on the column header, select Format Cells, in the item Number select Custom, and into the field "Type", type or cut and paste the date/time code as follows: mm/dd/yyyy hh:mm:ss .

c. Export the file to the clipboard. Click EDIT/PASTE TO CLIPBOARD. It can then be imported into any program that accepts data from the clipboard. You will usually want to specify tabs as the delimiter in the spreadsheet's main window before you export to the clipboard.

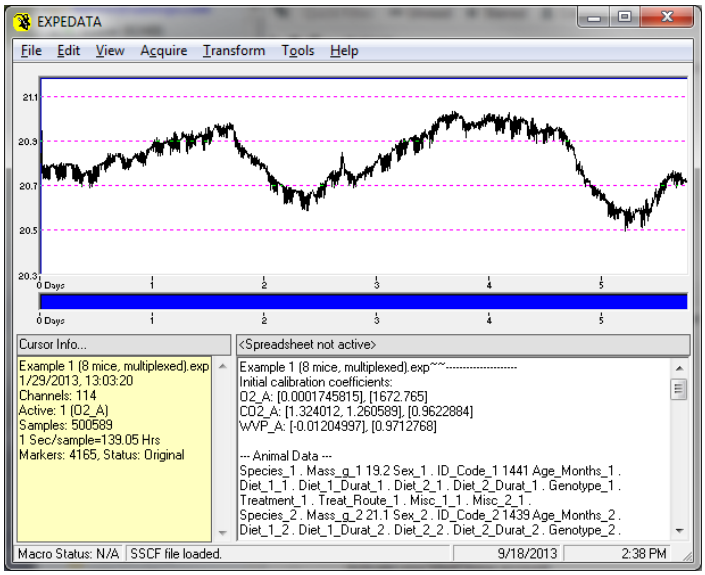
14. If you require data in another format, or if you have other requirements for extracted data, you may need a custom macro. In that case, you can either contact Sable Systems and arrange for it to be produced for you, or you can modify an existing macro or create your own. The advanced macro guide (at the end of this manual) will guide you if you wish to undertake this task yourself.

# BASIC DATA ANALYSIS 2: EXPLORING YOUR DATA IN MANUAL MODE

Although Expedata has a fairly steep learning curve if you use it to perform advanced data transformations or extractions, it is very simple to use for basic data visualization. This chapter will guide you through the basic process of data visualization.

## *Load the Data*

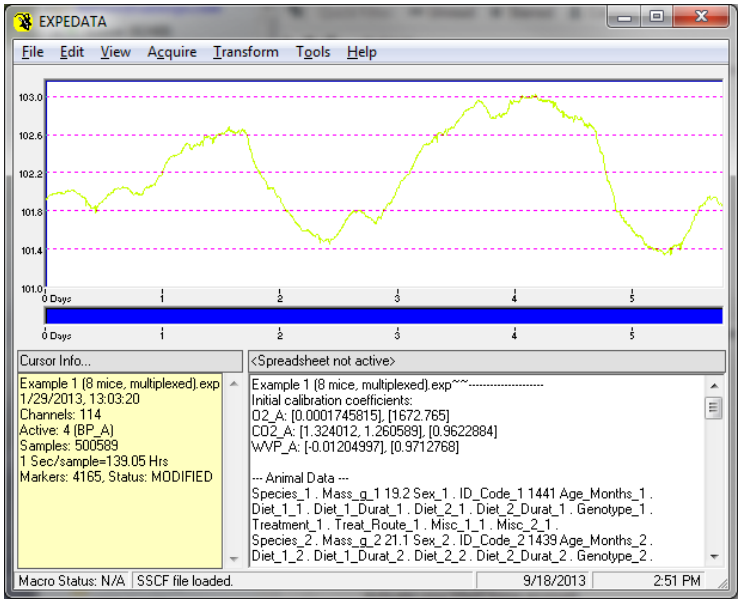
Start Expedata, click on file, and then click on open. Choose the data file, usually with the .exp extension, and click open. Something like the following window will appear.




In this example, we are looking at raw data from a multiplexed metabolic phenotyping system. This channel shows the raw oxygen data, which looks quite random and noisy. Actually it isn't - in this case, you are looking at 5 days of raw oxygen data, highly compressed (over 500,000 individual readings) and showing the influence of swings in ambient water vapor concentration and barometric pressure that will later be eliminated during data processing. Sometimes raw data isn't pretty.

## *Choose a Channel to Examine*

a typical data file will contain anywhere from 100 to over 400 channels of data, each acquired every second. a moment ago, we mentioned that the oxygen traces affected by barometric pressure. Let's see if this is actually the case. To do so, let's look at the barometric pressure during the recording. Click view, and then click active. The following dialog will appear. Scroll down to BP_A. In our example, here is the barometric pressure:

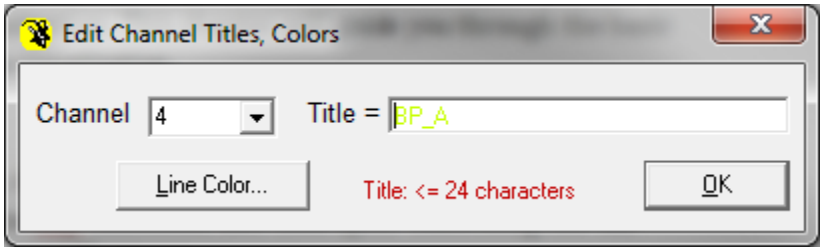



As you will notice, the oxygen trace, visible in the previous graph, faithfully tracks the barometric pressure signal. Obviously, if barometric pressure is not measured and is not compensated for, substantial inaccuracies can result.

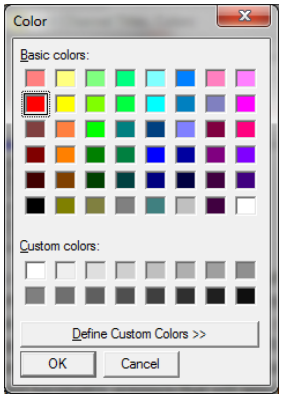
However, sometimes Expedata will choose a graphing color that is not very visible. Colors are automatically chosen to allow traces to be distinguished when graphed together (see later).

## *Change the Foreground Graph Color*

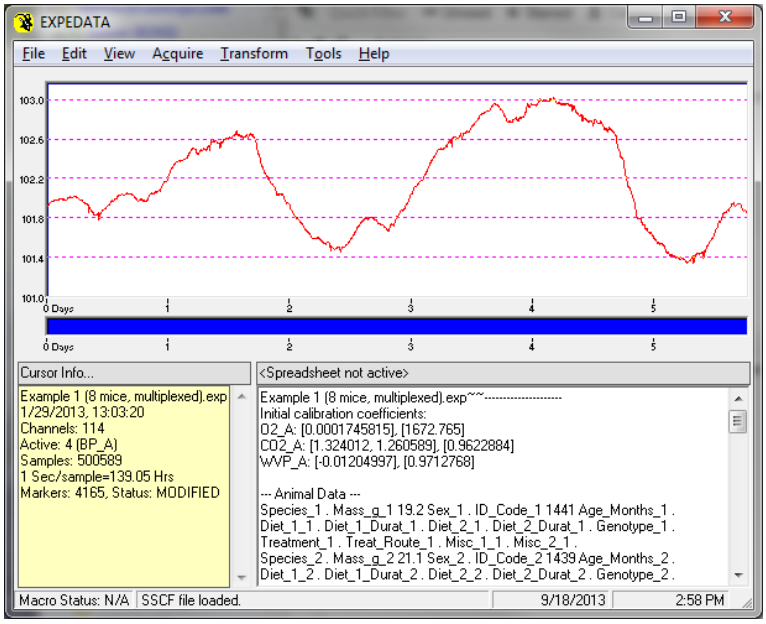
to change the displayed color of a graph, click on edit and then click on titles/colors. The following dialog will appear.




Click on the line color button. The color dialog will appear.

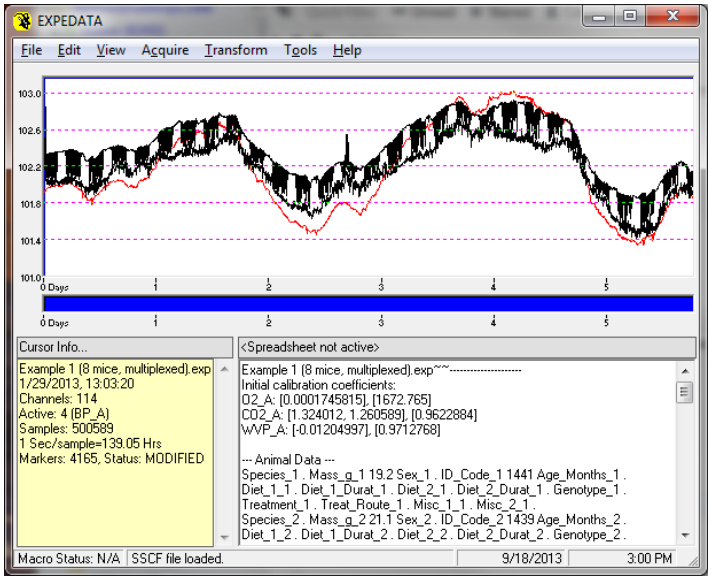



Click a nice visible color, then click okay and click okay again. The data will now be re-graphed in the color you have chosen (in this case red).




## *Display Multiple Channels at Once*

You can easily display multiple graphs on the same screen. To do so, click on view and then click display. Choose the data channel or channels that you would like to display, and click okay. In this example, the oxygen trace (O2_A) was clicked.




Notice that in this view, the oxygen trace is graphed in such a way that its upper and lower limits are shown. The upper limits correspond to baseline or incurrent air oxygen concentrations, while the lower limits correspond to oxygen concentrations in the air pulled from the different cages.

Note that the Y axis scaling applies only to the active channel. Displayed channels are automatically scaled to show all of their data, and are not necessarily to the same scale as the active channel.

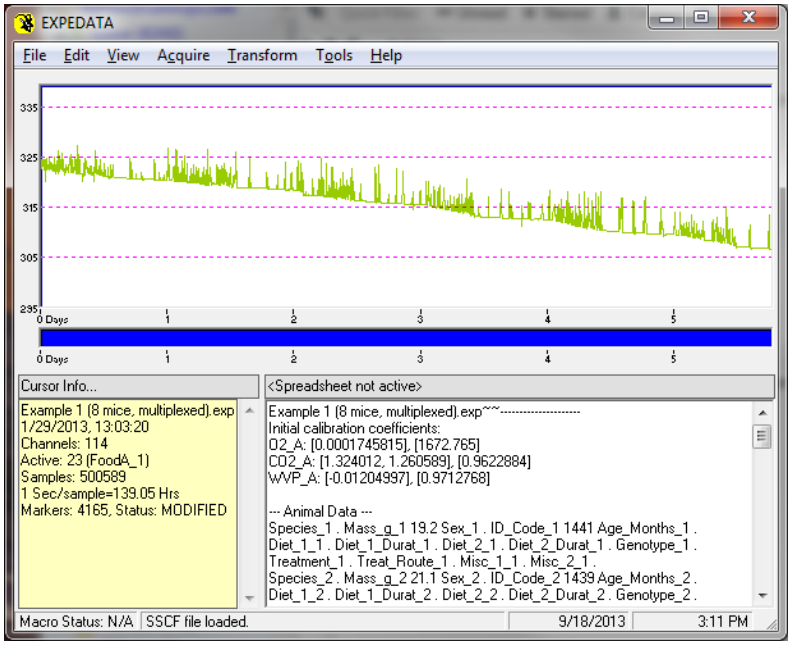
# What is the Date and Time at the Cursor?

As you move the mouse cursor over the graph, the cursor info window updates with the value of the active data channel. You can easily display the exact date and time of the cursor. To do so, hold down the CTRL key while moving the cursor. The cursor info window then updates the exact date and time corresponding to the cursor's position.

# Using raw data views for quality control

Channels of metabolic data undergo intense processing and automatically corrected in accordance with the instructions in the pre-processing macro, usually macro one in your macro collection. Other data channels, especially mass measurement channels, are not as self-correcting. If a mistake is made during setup, mass readings can be compromised to the point where accurate uptake or body mass measurements are impossible. Therefore, it is a good idea to scan quickly through the food uptake, water uptake and body mass channels to make sure that the data looks reasonable.

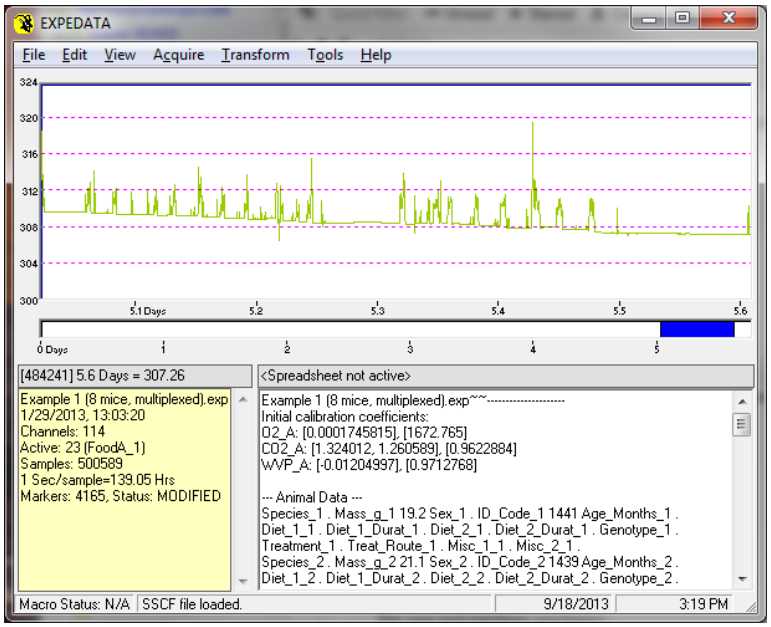
In our example, we will now make the food uptake sensor for the first cage (channel FoodA_1) active. (To turn off the display of other channels, click view, display, none, okay). Here is the result:




This is a very typical food uptake recording. Remember that you are looking at raw data, which in this case means the mass of the food Hopper during a five-day period. The spiky disturbances that you see are the animal feeding, exerting force on the copper. Let's zoom into the recording to examine it in more detail.

## *Zooming In*

To zoom into the recording, simply position the mouse cursor at the location you wish to zoom into, and then rotate the mouse wheel away from you. This is the result.



When food uptake data are analyzed, the stable mass values prior to and after each uptake event are compared. If the two values differ significantly, an uptake event is recorded. For more details, see the mass measurement chapter.

**To see where you are in the recording, look at the X axis, which shows you the number of hours that have elapsed since the start of the recording. Also, note the blue bar just below the graphed data. The blue section shows you where the displayed section is located in the entire recording, which is the width of the entire bar, white and blue sections combined.**

# Uptake data: quality control

Good uptake data shows a consistent downward trend across the length of the recording. Occasionally, you might find minor, slow upward excursions. These are caused by increases in the relative humidity within the cage, which cause the mass of the food to increase slightly. It is possible in principle to compensate for this effect, but doing so is beyond the scope of this chapter. Usually this effect is minor, and compensated for by the differential nature of uptake measurements, which are rapid compared to the much slower changes caused by water vapor absorption.

If you receive major and apparently random, rapid upward and downward excursions, this is a sign that the uptake data are compromised. Usually this is caused by something contacting the food or water hopper. That object might be bedding, or the animal inhabiting the cage. In either case, it is usually not possible to obtain accurate uptake measurements under the circumstances. They should be corrected as soon as possible. Contact Sable systems if in doubt.

# Displaying Related Channels

It can be rather tiresome, especially in long recordings with many cages, to display channels by selecting them from the view/active menu. A much faster method, used with related channels such as food uptake channels, water uptake channels etc., is to use a keyboard shortcut. Simply hold down the control and shift keys, and then use the left or right arrow keys to flip up or down through the cage number two display its selected graph. If you are also displaying related channels, these are automatically updated as well to show data from the cage in question.

# Selecting Data

To select a section of data, simply position the cursor at the start of the data section you want to select, click the left mouse button, and then drag it to the end of the data you want to select. The selected data window will appear, and a variety of options for examining the mean, area, slope, etc. of the selected data section will be available.

# Exporting Data

If you want to export data directly from Expedata interactive mode (which we are using here), it is simple to do so. Simply select the data section you wish to export, and then in the main Expedata window, click on file and

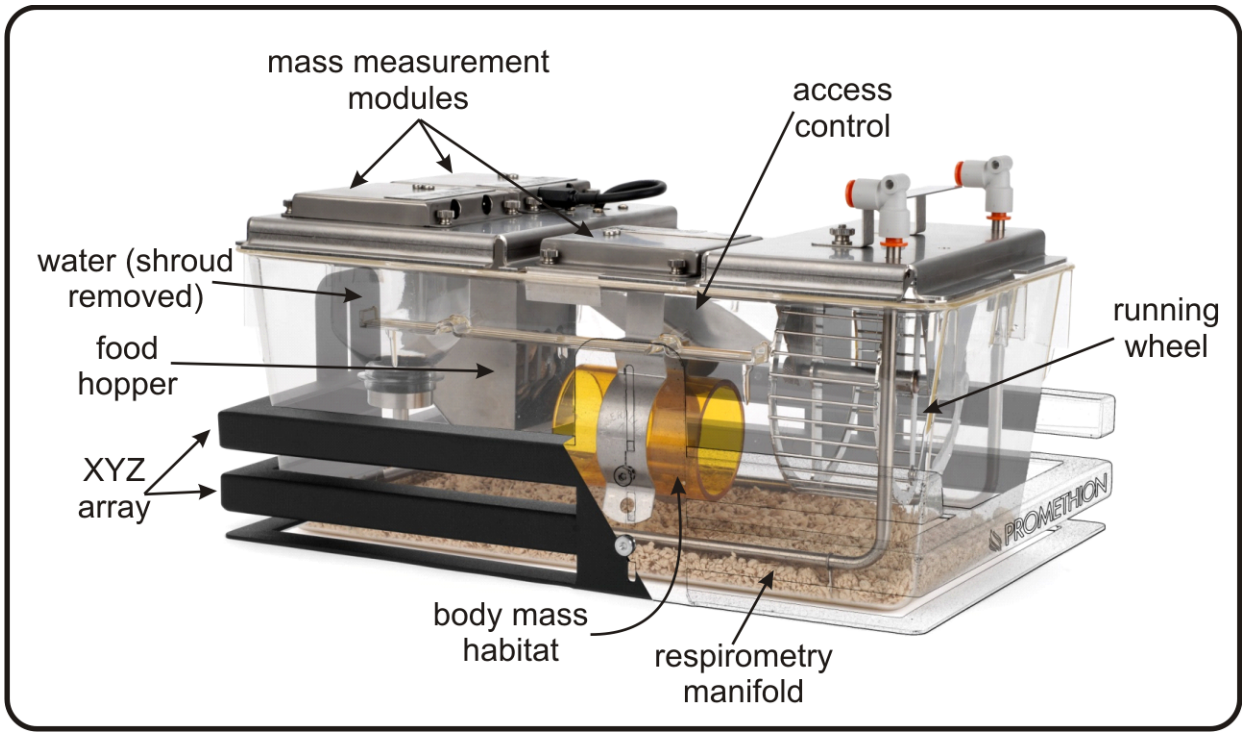
then click on export. Choose the channels you wish to export (by default, the active channel), and select the format and destination that you want to use. Simply follow the prompts. Be aware that exporting a substantial section of data may mean dealing with literally hundreds of thousands of data points. Fortunately, the data export dialog offers you the option of subsampling points to reduce the number of points that will be exported.

## Capturing a Quick Image

To capture an image of the Expedata window, for pasting into a document or for sending to Sable systems in case of a question, simply hold down the alt key and press the print screen key. This will place a graphic image of the Expedata window in the clipboard, from which it can be pasted into other applications such as word processors.

# THE CAGE AND ITS SENSORS: INTRODUCTION

The layout of a typical cage is shown below.




lk

# THE CAGE AND ITS SENSORS: MASS MEASUREMENT

## *Deploying And Calibrating MM-1 Mass Sensor Modules*

The Promethion MM-1 mass sensor module has a measurement range of 1 kg, and a resolution of 2-3 mg (SD of 30 second reading with averaging set to 2 seconds). It is intended for food and water uptake measurement and for body mass measurement. In both cases, the data acquired from it are in raw mass form, and are processed by algorithms in the acquisition (MetaScreen) or analysis (ExpeData) programs to yield the required data type. The MM-1 was designed to be as compact and unobtrusive as possible, allowing it to be used in the lids of cages without interfering with other measurement modalities such as respirometry.

*Caution*

The MM-1 is a precision weighing device and must be handled with care. When inserting or removing food or water hoppers or habitats (used for body mass measurement), always use the minimum amount of force possible. Also, be sure that its load platform (with hopper-mounting grooves) is firmly attached to the load cell with two Philips screws, and that its orientation is exactly square to the main body of the MM-1. If it is not, gently loosen the screws slightly, correct the orientation of the mounting plate, and gently tighten the screws. *It is very important that the load platform not touch the cage lid, or anything else other than the hopper or habitat attached to the load platform. Likewise, nothing must be allowed to contact the hopper or load platform within the cage, other than the mouse or rat, or accurate measurements will be impossible.*

*Deployment*

A typical deployment in a rodent cage uses three MM-1s. One is for food uptake measurement, one for water uptake measurement, and one for body mass measurement, or, optionally, for measuring food uptake from a second hopper. The MM-1s are all connected to a cage controller (CC-1) for that cage. Each MM-1 has a different network address; 1, 2 or 3. The Promethion system distinguishes between the MM-1s with different functions by assigning each MM-1 a network address that corresponds to its primary function. These are 1 for primary food uptake, 2 for water uptake, and 3 for body mass or secondary uptake (the function choice for address 3 is made in the MetaScreen program).

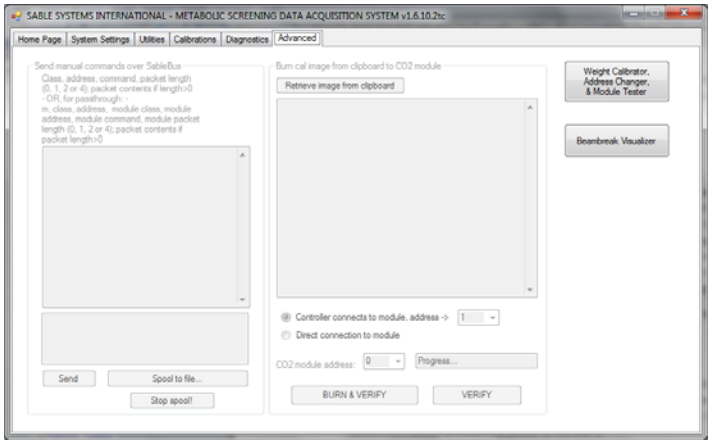
Therefore, you will need to assign an address to each MM-1 if that was not done at the time of installation of your system. You will also need to calibrate your MM-1s. For calibration, you will need a known mass. This is best achieved by filling a food hopper and weighing it on a lab balance. Aim at a mass of 200 – 700 g. The exact value is not critical but must be known. Note the mass and put the hopper within easy reach.

*Check on the address of your MM-1s*

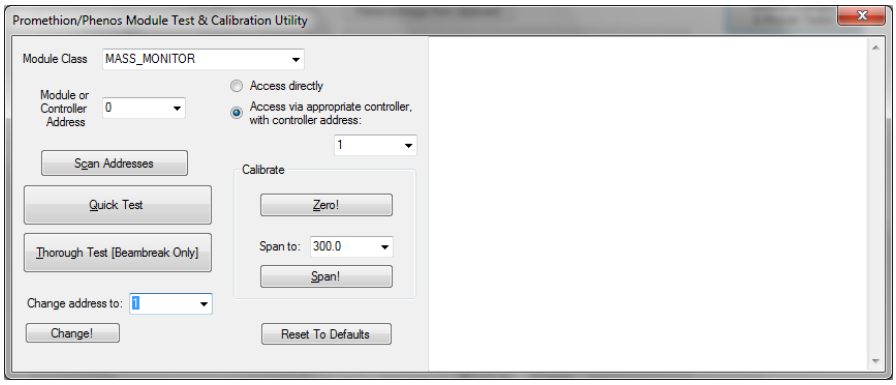
1. Connect the interface module (IM-1) to your computer using a USB to mini USB cord. Connect the IM-1 to a cage controller (CC-1) and power the CC-1 with its supplied power unit. Connect *only* the primary food uptake MM-1 to the CC-1 using a supplied TRRS cable. This supplies both a network connection and

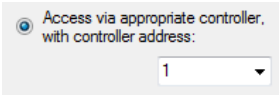
electrical power to the MM-1. The connected MM-1 should be mounted as usual in the cage lid. If you intend to calibrate your MM-1 (required at first deployment and recommended periodically thereafter), nothing should be mounted on the underside of the connected MM-1.

2. Log on to MetaScreen. Click on the ADVANCED tab. The following screen will appear.




3. Click on the WEIGHT CALIBRATOR AND ADDRESS CHANGER button. The following dialog will appear.




4. Enter the cage controller (CC-1) address 

. The cage controller's address should be the same as the number of the cage with which it is associated.

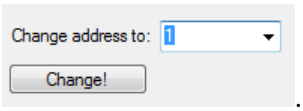
5. Click on SCAN ADDRESSES 

. If the program cannot find the cage controller it will return an error. In that case, the cage controller address is wrong, or the cage controller is not powered. Remedy the problem before proceeding.

6. If the scan was successful, the text box to the right of the screen will display "Module MASS_MONITOR found at address" followed by the address number of the connected module. The selected MODULE ADDRESS text value will change to that address, which will be used for subsequent communication

unless changed. That address should be 1 for a primary food uptake MM-1, 2 for a water uptake MM-1 and 3 for a body mass or secondary uptake MM-1.

7. If the returned address does not match the function of the MM-1, it must be changed. We suggest labeling your MM-1s with their addresses so that this only needs to be done once.

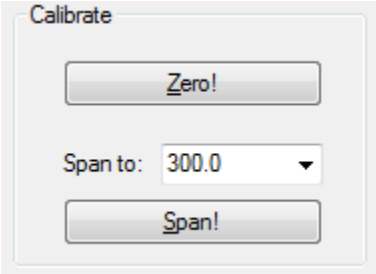
8. *To change an address,* enter the new address and click CHANGE 

. The address will be changed and communication tested.

9. Click on QUICK TEST. The text box to the right of the screen will display the current mass measured by the MM-1.

10. Calibration is required at first deployment and periodically thereafter.

11. *To calibrate an MM-1,* it must first be zeroed, then spanned. These operations are carried out in the CALIBRATE group of controls.




12. *To zero an MM-1,* make sure that nothing is attached to its load platform. Then, check that it is giving stable readings by repeatedly clicking on QUICK TEST. The mass readings should vary by 0.01-0.02 g or less from reading to reading. The MM-1 may drift by a few tenths of a gram from first turn-on to the point when its mass readings stabilize. When the reading is stable, click on ZERO!. After zeroing an MM-1 at first deployment, it *must* then be spanned.

13. *To span an MM-1*, first zero it with nothing attached, and mounted in its usual position. Then attach the weighed food hopper or other accurately known mass. Enter that mass in grams into the SPAN TO box. Check that the MM-1 is giving stable readings by repeatedly clicking on QUICK TEST. When the reading has stabilized, click on SPAN!.

14. We recommend re-checking zero and span if this is the MM-1's first deployment.

15. Repeat the above steps as required for the other mass sensors, addressing them according to their station in life, and calibrating them. At first deployment, only allow one MM-1 to be connected to the cage controller at a time until all have been addressed as required. At that point, you can connect them

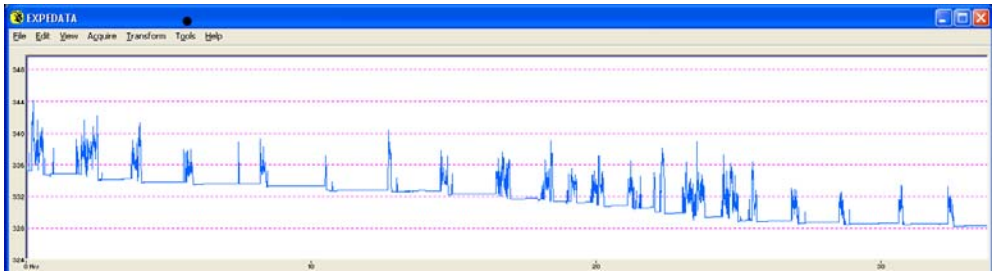
all, and change the 

as required to interrogate or calibrate any MM-1 attached to the selected cage controller.

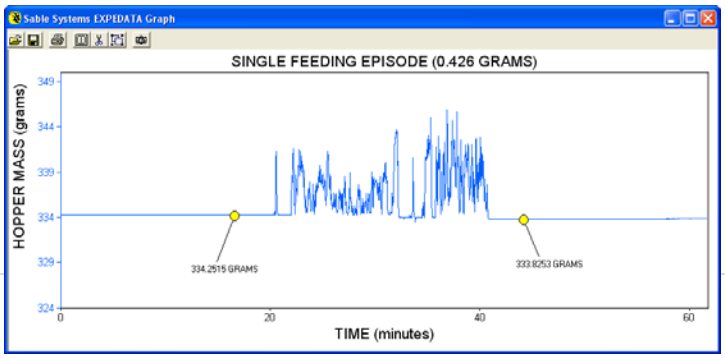
## *Uptake Analysis*

The Promethion system continuously monitors the masses of all food hoppers and water dispensers attached to each cage. By default, the mass readings are stored every second, so that a continuous record of food or water uptake is maintained. This allows practically every aspect of uptake behavior to be recorded; you can extract any statistical data you wish from the raw data in the recordings. Moreover, the raw data are future-proof. If another analytical protocol is developed in the future, or if unanticipated aspects of feeding behavior are later recognized as important, you can re-analyze the raw data with new goals in mind that were not anticipated when the recording was made. Systems that record only condensed, processed data do not allow this flexibility, and also do not allow the user to evaluate how effectively the uptake calculations are, in fact, operating.

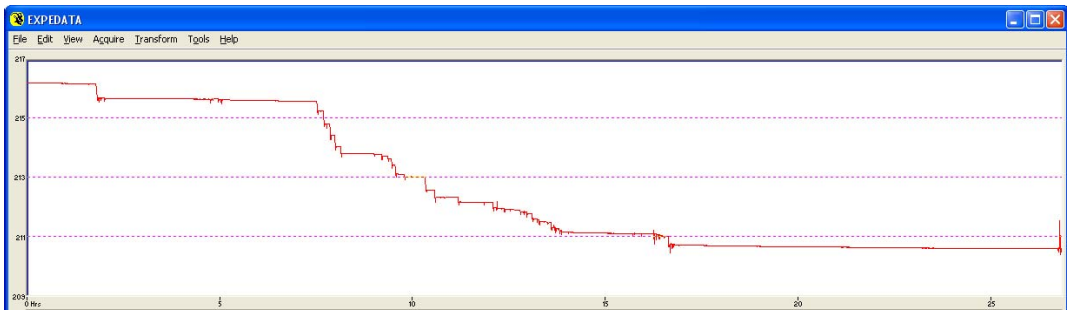
When an animal eats from a hopper or drinks from a water dispenser, it disrupts the mass recording. Food and water uptake (or intake) analysis depends on recognizing when this disruption occurs, and calculating the difference in stable masses measured just before, and just after, the disruption. Unlike naive algorithms that simply weight the hopper and store stable values relative to the start of the monitoring session, this technique is insensitive to slow changes in mass that may occur and that are unrelated to food uptake. Such changes are usually caused by temperature swings (all weighing sensors are sensitive to temperature), or humidity fluctuations (the mass of rodent chow changes according to its water content, which is a function of ambient humidity). The following graph (Fig. 1) shows a 34-hour feeding record from a mouse.



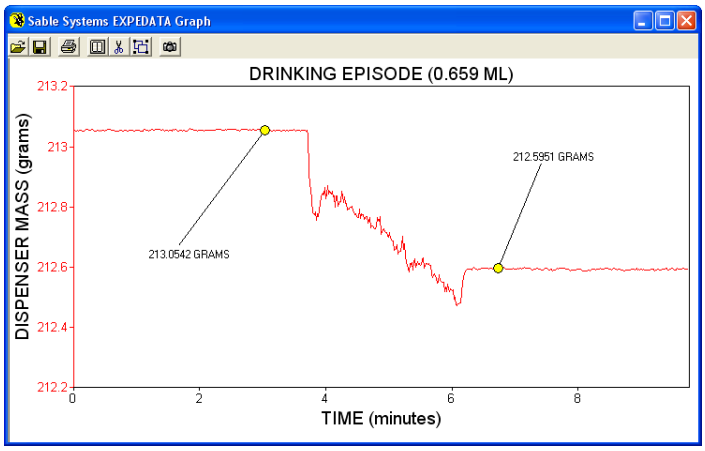

Zooming in on a single feeding episode, we see (Fig. 2):



Here is a water uptake record (Fig. 3):




And, as with food uptake, zooming in on a single drinking episode, we see (Fig. 4):




Although the feeding and drinking records look very different, they have one feature on common: During food or water uptake, the mass readings are consistently highly variable when compared to the quiescent values measured when the animal is not interacting with the dispenser. By obtaining stable mass readings before and after the uptake event, the exact amount of material removed during that event is precisely measured. A wealth of additional information, not currently used in scientific publications because no other system offers this depth of detail, is also available.

Note that we have to distinguish between uptake and quiescent states based on the variability of the measured masses, or we cannot measure uptake amounts. Thus, for any food or water uptake algorithm to work, noise in the weighing sensor must be minimized. The load cell sensor used by Promethion systems is equivalent to those used in precision balances, and is capable of recording mass with a resolution of ~2 mg SD over a 1 kg range. That is ~1 part in 500,000. This is more than adequate for the job. However, some slight care is required to ensure that the load sensor's hopper or dispenser holder does not touch the cage lid. If it does touch the cage lid, the mass recording will be about 10 – 100 times more noisy than it should be, and it may become difficult or impossible to distinguish between uptake events and the quiescent, non-contacted state of the load sensor. If

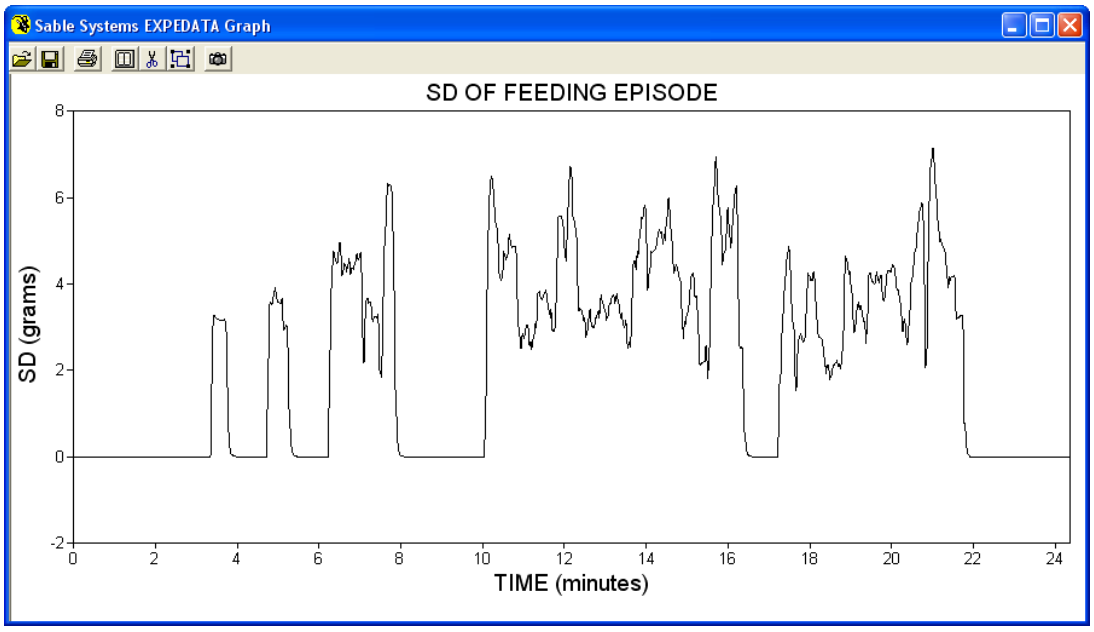
this is the case, accuracy will suffer. Always check to ensure that the load sensor's hopper or dispenser holder is true and square so that it does not touch the cage lid, and that any retention mechanisms that prevent the hopper or dispenser from moving are engaged. Also ensure that the access control system (if used) does not impinge on the hopper holder.

A wide variety of analytical protocols can be employed to extract uptake data from raw mass data. After extensive experimentation we have developed a flexible analytical scheme that is applicable to practically any uptake scenario.

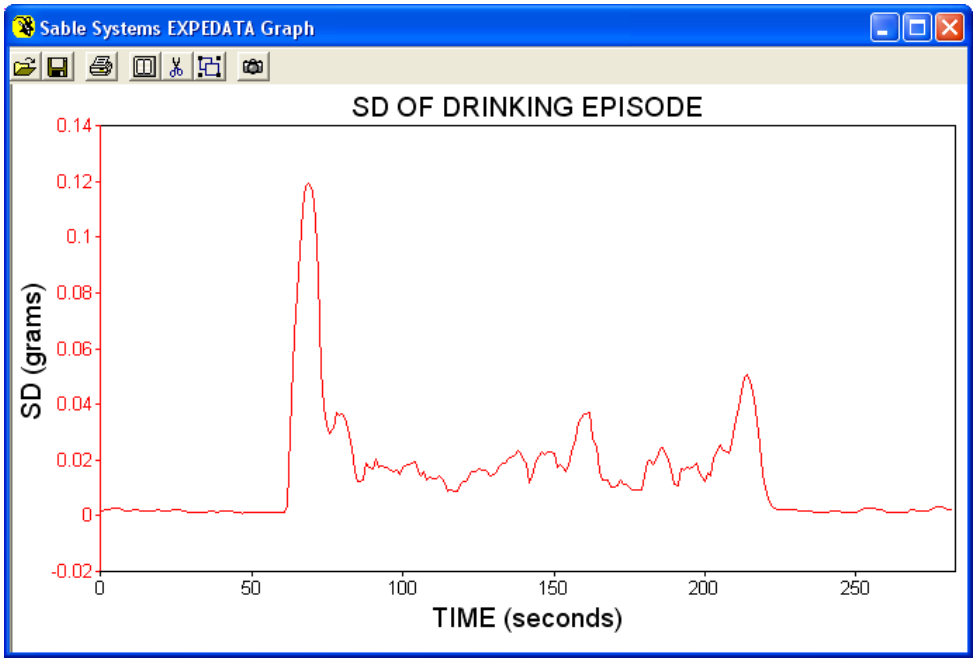
We will now explain how this scheme (or algorithm, if you prefer) works. We will be happy to analyze your raw data and provide you with customized macros that extract data to your specifications, in which case you can get away with not understanding what is happening behind the scenes. However, we urge you to understand how this scheme works, because (a) doing so will allow you to write and modify your own macros, and (b) you should in principle understand how data on which you will rely has been produced.

The first step in the algorithm is to produce a vector of mass variability. This is simply a vector (or channel, if you prefer) that is created by calculating the standard deviation of the mass channel. For this to be useful, the standard deviation (SD) is calculated over a sliding window (typically 3 – 20 seconds in width) that proceeds from the beginning to the end of the trace, moving one second at a time and storing the SD at each point. The result is a vector that contains the *variability* of the data rather than its absolute value. Within ExpeData, you can easily apply a sliding SD transform (TRANSFORM/EXOTIC TRANSFORMS/SLIDING STD DEVIATION), and we urge you to try this on your food and water uptake data. Use a sliding window of 5 samples for food uptake, and 10 samples for water uptake. After examining the result, undo the transformation with EDIT/UNDO.

Here is an example of a feeding episode transformed by a 20-sample sliding SD (Fig. 5):



And, as with food uptake SD, zooming in on a single drinking episode, we see (Fig. 6):




As you can see, uptake events are easily identifiable by visual inspection of the graphs. The uptake quantification algorithm makes use of a sliding SD to distinguish uptake episodes from the quiescent state of the load sensor. (You do not need to create the SD vector, as shown on the previous page so that you could visualize the process; it is created automatically, behind the scenes).

Obviously, the algorithm must not be fooled by events that are not, in fact, uptake events. These might include brief, non-uptake contact, or mechanically-induced noise from various sources. Fortunately, uptake events are fairly distinctive. They are not momentary, but last for some minimum interval, though pauses may occur within that interval; and they change the mass of the hopper or dispenser. Thus, to produce meaningful data, uptake events must meet five vital characteristics:

1. The SD associated with the event must exceed a user-defined critical value. This defines the start of the uptake event.

2. The SD associated with the event must exceed that critical value for at least a user-defined minimum time interval. A return to quiescent values defines the end of the uptake event.

3. Pauses in uptake within the uptake event are permitted, but must not exceed a user-defined maximum pause duration. If they do, the uptake event after the pause is treated as a separate event.

4. The overall SD fluctuation from the start to the end of the uptake event must exceed one-quarter of the critical SD. This rejects brief, repeated contact events.

5. The mass readings before and after the event must differ at a user-defined significance level; the default is $P \leq 0.05$, but you can set it lower (fussier) or higher (more forgiving). The probability is automatically determined using Student's *t*-test.

6. You can set the minimum food uptake mass that counts as a food uptake event. By default, this value is 0 ("uptake events" that *add* mass to the food hopper are not counted as uptake events). If you want to include such events, set that parameter to a negative number. If you want to reject uptake events of less than (for example) 0.1 g, then set that parameter to 0.1.

Between them, these parameters allow an uptake episode (whether feeding or drinking) to be recognized reliably, and the accompanying uptake amounts to be measured accurately from the stable mass readings recorded before and after the event. Unlike other uptake measurement systems, Promethion systems record the raw data, so that the correct operation of the uptake algorithm can be rigorously validated. Systems that record only processed data cannot be validated in this way; the manufacturer's canned algorithms must be trusted instead, and even if they are correct for a given strain of a given study animal, they cannot be fine-tuned to meet challenges caused by differences between species such as rats *vs.* mice, or between strains within species.

Fortunately, the Promethion uptake determination algorithm is robust enough to require little or no fine-tuning across a wide range of uptake behaviors. However, it is usually necessary to use different iterations of the algorithm – that is, with some of the above parameters set differently – when determining food uptake, or water uptake.

The algorithm is not limited to producing cumulative food uptake graphs. It can also produce detailed uptake reports which include uptake duration, uptake amount, and statistical details for each update event. Moreover, the algorithm can optionally group individual uptake events that occur close together in time as "meals". For example, a "meal" can be defined as a series of uptake events separated by no more than 15 minutes, and having a total summed uptake of over 0.2 g. These parameters are easy to specify in the uptake analysis dialog.

If you require additional detail beyond this, please send us a detailed request, and if it is feasible, we will add the additional requested analytical features during the next development cycle. Below is an example of the default detailed data output, which is saved in the same folder as the recording, with the channel name (e.g. FOODA_1) and, where appropriate, the selection area, appended to the recording name. The file is in standard text format (.txt) for easy importing.

| Event | SDate | STime | EDate | ETime | Minutes | Grams | G_min | Stud_t | Prob |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 9/9/2010 | 16:15:59 | 9/9/2010 | 16:23:16 | 7.3 | 1.696 | 0.232 | 82.02 | 0 |
| 2 | 9/9/2010 | 18:47:59 | 9/9/2010 | 18:54:02 | 6.07 | 1.007 | 0.166 | 51.04 | 0 |
| 3 | 9/9/2010 | 19:12:30 | 9/9/2010 | 19:23:27 | 10.97 | 3.023 | 0.276 | 145.54 | 0 |
| 4 | 9/9/2010 | 20:58:36 | 9/9/2010 | 21:09:28 | 10.88 | 1.27 | 0.117 | 48.42 | 0 |
| 5 | 9/9/2010 | 21:33:29 | 9/9/2010 | 21:42:09 | 8.68 | 2.549 | 0.294 | 142.55 | 0 |
| 6 | 9/9/2010 | 23:03:02 | 9/9/2010 | 23:05:23 | 2.37 | 0.399 | 0.169 | 13.45 | 0 |
| 7 | 9/9/2010 | 23:57:14 | 9/9/2010 | 23:59:29 | 2.27 | 0.957 | 0.422 | 22.55 | 0 |
| 8 | 9/10/2010 | 0:01:23 | 9/10/2010 | 0:04:20 | 2.97 | 0.384 | 0.129 | 19.7 | 0 |
| 9 | 9/10/2010 | 0:53:13 | 9/10/2010 | 1:08:05 | 14.88 | 3.403 | 0.229 | 35.7 | 0 |
| 10 | 9/10/2010 | 3:25:28 | 9/10/2010 | 3:37:44 | 12.28 | 2.49 | 0.203 | 87.92 | 0 |
| 11 | 9/10/2010 | 4:20:15 | 9/10/2010 | 4:25:19 | 5.08 | 1.146 | 0.225 | 62.24 | 0 |
| 12 | 9/10/2010 | 4:35:16 | 9/10/2010 | 4:47:24 | 12.15 | 3.748 | 0.308 | 35.82 | 0 |
| 13 | 9/10/2010 | 8:58:22 | 9/10/2010 | 9:14:50 | 16.48 | 3.6 | 0.218 | 12.69 | 0 |
| | | | | | | | | | |

The easiest way to proceed at this point is to send us some test recordings of food and water uptake from the strains you are using. We can then determine and test the macro command to yield the data you require. However, the process is quite simple, and you can save some time by following these instructions and trying it yourself. We consider food and water uptake separately.

*Food uptake*

Open the file in ExpeData and follow these steps:

1. Make a typical food uptake channel active (VIEW/ACTIVE)
2. Apply the sliding SD transform with a 5 second window (TRANSFORM/EXOTIC TRANSFORMS/SLIDING STD DEVIATION).
3. Examine the data. Zoom in or out as required.
4. Determine a typical minimum SD value that marks the start of a feeding bout. For most animals, an SD value of 0.02 to 0.05 gram will work well for food uptake.
5. Determine the minimum duration of a feeding bout, and how long any interruptions within bouts may last without being categorized as a separate bout. For most purposes 45 and 60 seconds will be adequate, but you can change these values based on how you wish to define feeding bouts.
6. Undo the SD transform (EDIT/UNDO). Turn on macro recording, if applicable. Apply the uptake transform (TRANSFORM/EXOTIC TRANSFORMS/UPTAKE PROCESSING), filling in the above values as prompted.
7. If you want a detailed breakdown of the results saved as a separate file, check that option.

8. If you are creating a macro, it is quickest to transform one uptake channel while recording the macro, then stop the macro and edit it to create the complete macro for all of the channels, preferably using a looping construct (see *Data Analysis*).

The result is a channel showing cumulative food uptake amounts.

*Water uptake*

1. Make a typical water uptake channel active (VIEW/ACTIVE).
2. Apply the sliding SD transform with a 5 second window (TRANSFORM/EXOTIC TRANSFORMS/SLIDING STD DEVIATON).
3. Examine the data. Zoom in or out as required.
4. Determine a typical minimum SD value that marks the start of a drinking bout. For most animals, an SD value of 0.02 gram will work for water uptake.
5. Determine the minimum duration of a drinking bout, and how long any interruptions within bouts may last. For most purposes 15 and 15 seconds will be adequate, but you can change these values based on how you wish to define drinking bouts.
6. Undo the SD transform (EDIT/UNDO). Turn on macro recording, if applicable. Apply the uptake transform (TRANSFORM/EXOTIC TRANSFORMS/UPTAKE PROCESSING), filling in the above values as prompted.
7. If you want a detailed breakdown of the results saved as a separate file, check that option.
8. If you are creating a macro, it is quickest to transform one uptake channel while recording the macro, then stop the macro and edit it to create the complete macro for all of the channels, preferably using a looping construct (see *Data Analysis*).

The result is a channel showing cumulative water uptake amounts.

*Unexpected results?*

The data you can extract from a mass record are dependent on the quality of the mass record itself. The following issues can create results that may be meaningless:

1. Contact, no matter how slight, between the dispenser holder and the cage top or the uptake controller (if applicable).
2. Mounting the load sensor wrongly, so that the dispenser holder is actually pinched against the cage top.
3. Routine and prolonged non-uptake contact between the animal and the load sensor. Food dispenser shields can reduce this problem.

4. Significant temperature fluctuations, e.g. from changing a temperature cabinet's setpoint.

Usually, the nature of the problem will be quite obvious from the raw data recording. If in doubt, please contact us and we will be happy to assist in interpretation.

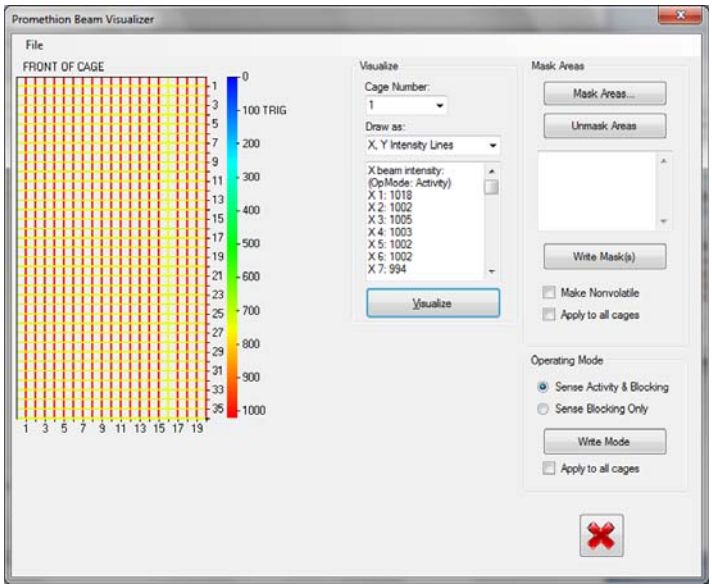
# THE CAGE AND ITS SENSORS: ACTIVITY MEASUREMENT

## *The XY(Z) Array*

The XYZ array consists of three independent, high resolution (1 cm spacing) infrared beam arrays that are invisible to all known animals (wavelength ca. 900 nm; mammals can only see wavelengths below about 750 nm). The intensities of these beams are measured in rapid succession. Obstacles such as experimental animals are easily detected because they cause the intensity of the beam to decline to near zero. An intelligent centroid-algorithm calculates the centroid of the animal, allowing the position of such obstacles to be accurately determined.

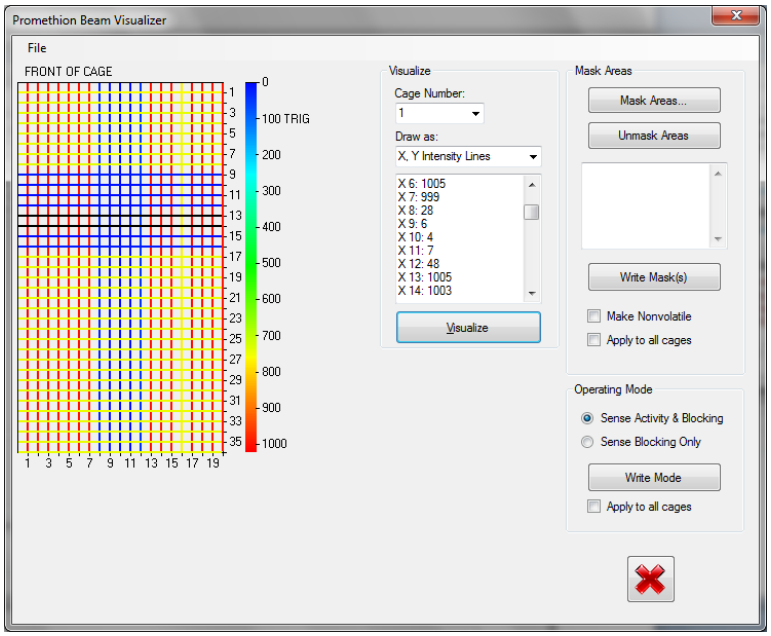
## Initial Testing

Start with an empty XYZ array (with no cage) connected to a free cage controller to get a feel for how the beambreak system works in practice. In MetaScreen, click on the ADVANCED tab and then click on BEAMBREAK VISUALIZER. The Promethion beambreak visualizer window will appear. Enter your cage controller number (default is 0), and click VISUALIZE. Your window will look something like this:




Note the crosshatch in the X and Y directions. Each line in the crosshatch corresponds to an infrared beam. The intensity of the beam, as sensed across the array of beams, is color-coded. The higher the intensity of the beam, the more the color of the line representing each beam moves from black, through blue, green, yellow and red. Black represents a beam that is completely blocked; red or yellow, a beam that has not encountered any meaningful obstacles. Each beam has a numerical value denoting its intensity, with 0 being the minimum possible intensity and 1023 being the maximum. The actual values are listed in a scrollable window. The relation between the numerical values and the color code is displayed on the vertical color bar next to the beambreak matrix. The color code is meant only as a handy diagnostic tool that aids in setting up the array, as you will see shortly.

You will notice that the color bar has the word “TRIG” near the top, opposite the value 100. This means that Promethion considers any beam with a measured intensity below 100 to be blocked.




To see this in action, place an opaque object in the beambreak array. As you can see, the object is detected. Note the beam intensity values in the scrollable text widow. Where the object is present, all values are well below the trigger intensity of 100.

Internally, the beam break system calculates the centroid of the object to a precision of ¼ of the inter-beam spacing. The inter-beam spacing is 1 cm, so the centroid precision is nominally 2.5 mm.

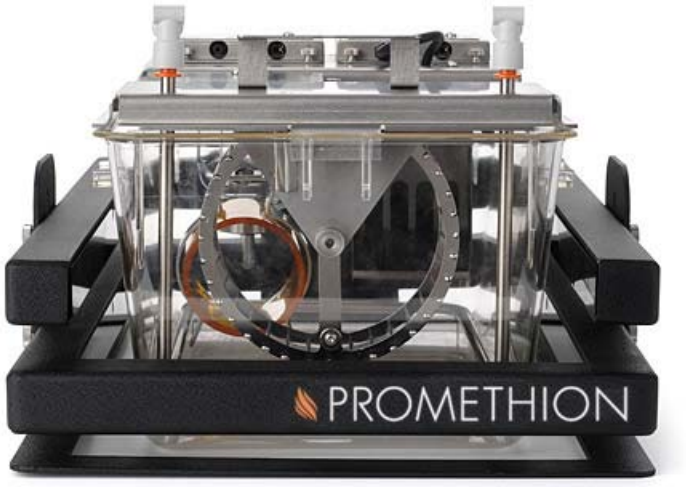
It is obvious that objects such as mice or rats can only be detected if they are in the path of a beam of reasonable intensity, certainly > 100. If another object, such as the respirometry manifold, a food hopper, or a pile of bedding obscures the beam, then that beam cannot be used to derive any useful information.

In practice, therefore, the height of the XY array and of the Z array must be carefully adjusted so that beam blockage is reduced to the minimum.

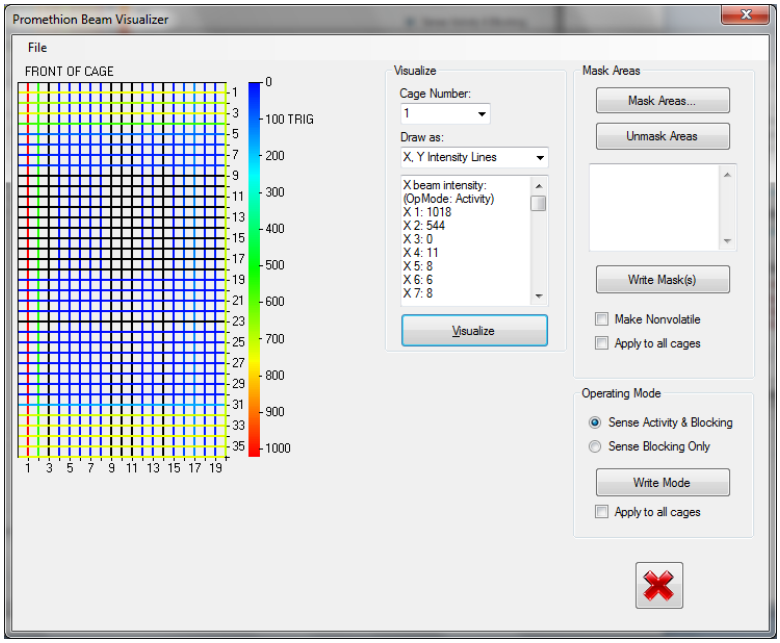
If a beambreak array is inexplicably “blind” in some area, it has probably been masked there (masks are explained below) and the mask has been made non-volatile. To undo this, simply click on UNMASK AREAS (you may need to click on MASK AREAS first to make the UNMASK button active), check the MAKE NONVOLATILE box and click on WRITE MASK(S).

## Setting up the XY array

Now let us place a cage in the XY beambreak array, in this orientation.

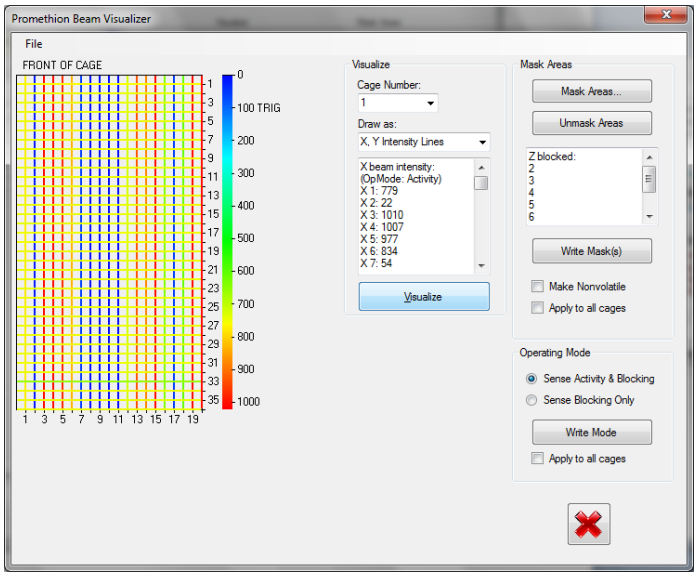



Position the cage so that its front (with the running wheel foremost) is nestled in the front left of the beam break array (to your left when looking at the picture above). This gives all cages consistent coordinates. Adjust the XY array to its lowest position. This may be the result when you click VISUALIZE:

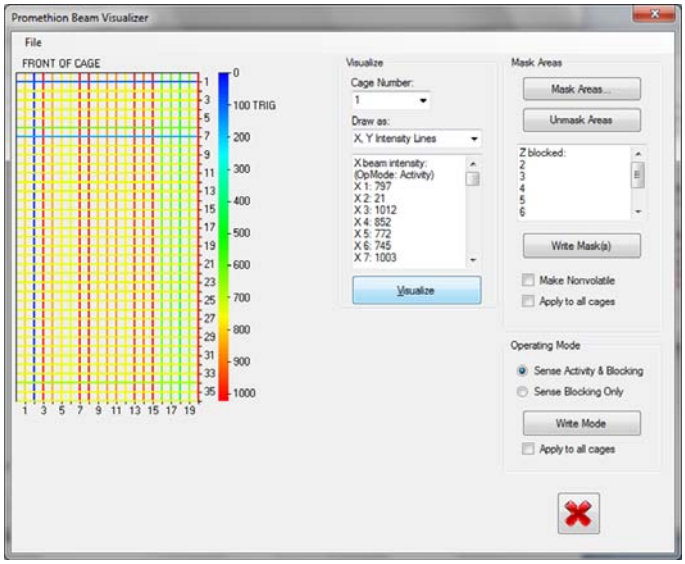



As you can see, many of the beams are blocked (blue or black = blocked, yellow or orange = not blocked). If the array is used in this state, it will not give us precise location information. In addition, it will not give us beam-break counts that are meaningful. This is because a blocked beam is already broken, and if the animal's activity is to be measured, the beam must be intact until the animal breaks it. Especially if you want to compare beam-break counts between cages, you will need to ensure that all of the beambreak arrays for all cages are properly adjusted to minimize blocked beams when no animal is present.

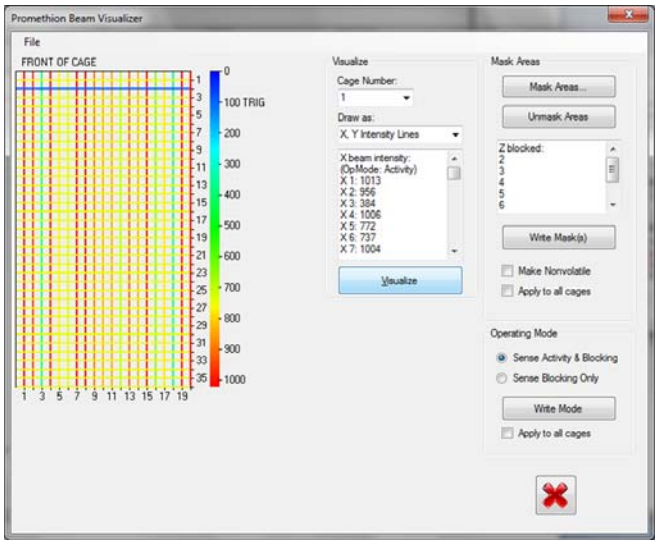
In this case the XY array is too low. Let's raise it slightly. Depending on your array, the adjustment technique will differ slightly. In older Promethion systems, three thumbscrews allow you to set the XY array's height. In newer systems, the beambreak array's mounting bracket has notches that allow the array to be quickly positioned.




That's much better. All of the Y beams have good intensity, though a few X beams are still blocked. A slight upward adjustment yields this:




We're almost there. Note the two blocked beams in the Y axis, near the front of the cage. These are caused by the front and rear rims of the running wheel. So, our array is a touch too high. A slight reduction in height yields this result:

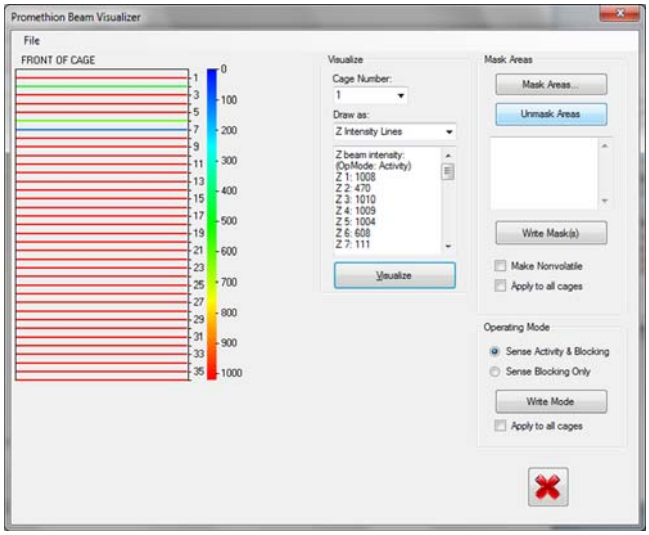



This is excellent. The respirometry manifold tubing run obscures one beam, but we can live with a blocked beam or two because the beambreak array adapts intelligently to minor blockages.

This adjustment must be made for each cage. With beambreak arrays that are positioned using thumbscrews, it can help to have blocks of an appropriate height to slip beneath the arrays. With the notched beambreak mounts, positioning is easier. If you are using a notched beambreak mount and determine that an intermediate position (between two notches) would work best, simply position the beambreak array at the higher notch position, and then place a flat object approximately as large as the cage's footprint beneath the cage to raise it slightly. The notch spacing is 11 mm; a sheet of 20 lb paper is typically 0.1 mm thick, so a few sheets of printer or copier paper provide a handy and repeatable spacer if this degree of fine adjustment is desired. Once the beambreak array for one cage has been set up, the adjustment of the other arrays will be similar or identical.

## Setting up the Z axis array

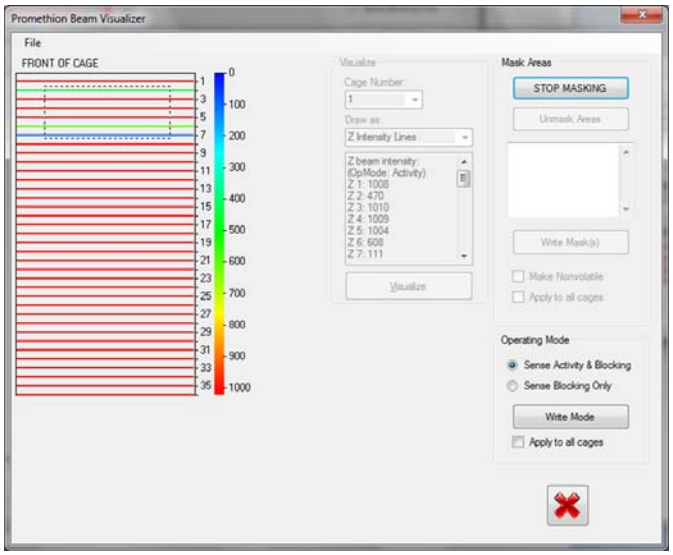
To set up the Z axis array, select Z INTENSITY LINES from the "Draw As" drop-down menu and click VISUALIZE. Follow the above procedure. The position of the Z axis beambreak array is far less critical that that of the XY array. It should be high enough to be blocked only when the animal rears, not when it is merely walking around. Some beam blockage is inevitable on the Z axis; in all cases the food and water hoppers will obscure some beams, as will the running wheel if one is installed.




Here is an example. The Z axis is correctly positioned. Note that for illustrative purposes, this cage does not have a food or water hopper. If they were present, the array would display some completely blocked beams towards the back of the cage (bottom of this figure).

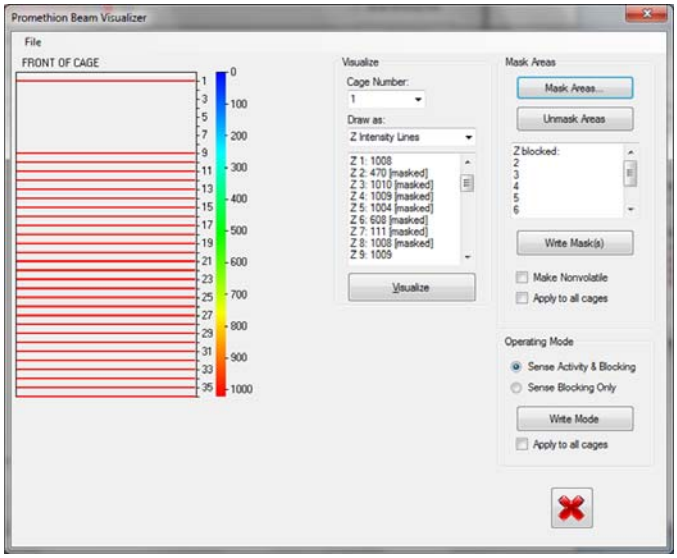
There is a problem here. The running wheel is visible (it blocks beams) and its motion, and/or the movements of the mouse within it, will be counted as Z axis activity. Typically the Z axis is used to give an indication of when the animal rears onto its hind feet, so counting beam-breaks caused by the running wheel and its occupant is not desirable. We need to *mask* the location of the running wheel. *Masking* beams means that the beambreak array ignores beams within a certain, settable area. How is this done?

Click MASK AREAS. Draw a rectangle in the area you wish to mask. In the case of the Z axis, only the height of the rectangle is relevant. In this case, we are masking the location of the running wheel.



In practice it is also a good idea to mask the food and water hoppers (not shown here), being careful not to extend past them and mask any unblocked beams. You can mask any number of areas, which will be combined when you click STOP MASKING.

If you make a mistake, simply click STOP MASKING, then UNMASK AREAS. This will erase any masks drawn for this beam-break array. You can then draw them again.




The running wheel is now masked, so that running activity will not be counted as rearing behavior. To undo the mask, click on UNMASK AREAS. To keep this setting for future runs using this beambreak array, check the MAKE NONVOLATILE box. If you are confident that all cages will be placed in exactly the same position within the beam-break array (positioned with the front of the cage in the front left corner of the beam-break array as we recommend), check the APPLY TO ALL CAGES box, otherwise this mask will only be applied to this cage. Then click on WRITE MASK(S).

Cage masking data (if any) are stored in your setup files and are automatically uploaded to the beambreak arrays at the start of each recording. So, if you set masks, be sure to save your setup file afterwards. Be aware that if the beambreaks are not make nonvolatile, then if a power failure occurs, the mask data will be lost, so in general it is a good idea to make them nonvolatile. To do so, you will need to set the masks as described above for each beam-break array in your system. The WRITE button only writes to the currently selected cage's beambreak arrays.

If a beambreak array is inexplicably "blind" in some area, it has probably been masked there and the mask has been made non-volatile. To undo this, simply click on UNMASK AREAS (you may need to click on MASK AREAS first to make the UNMASK button active), check the MAKE NONVOLATILE box and click on WRITE MASK(S).

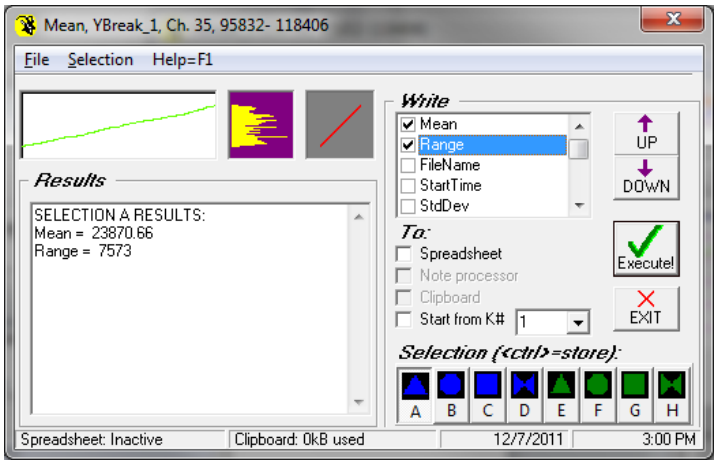
We recommend leaving the operating mode on SENSE ACTIVITY AND BLOCKING. This allows the beam-break array to slowly become "blind" to persistent obstacles.

## Data Analysis

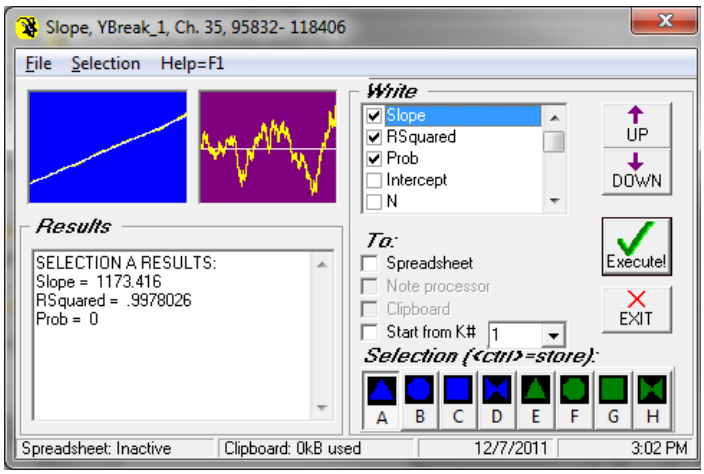
*Beambreak Counts*

By far the most common analytical use for XYZ array data is beambreak counts. These are simply the sum of occasions on which any beam on a given axis has been broken. The animal can't move around the cage without breaking beams if the beambreak arrays have been correctly set up, and thus the sum of breaks provides a handy operational index of overall activity. For this purpose the Y axis is most frequently used. The Z axis provides an operational index of rearing events. The most robust treatments of beambreak counts within a selected interval are the range (available as an option when taking the mean of the area) over that interval, or the slope of the cumulative beambreak trace vs. time in (for example) minutes or hours.

Here is the range of a beambreak trace over a selected section (the mean is shown too):




This says that the Y beam was broken 7573 times in the selected section. And here is the slope of the line over the same section: (The purple window shows the residuals of the regression).



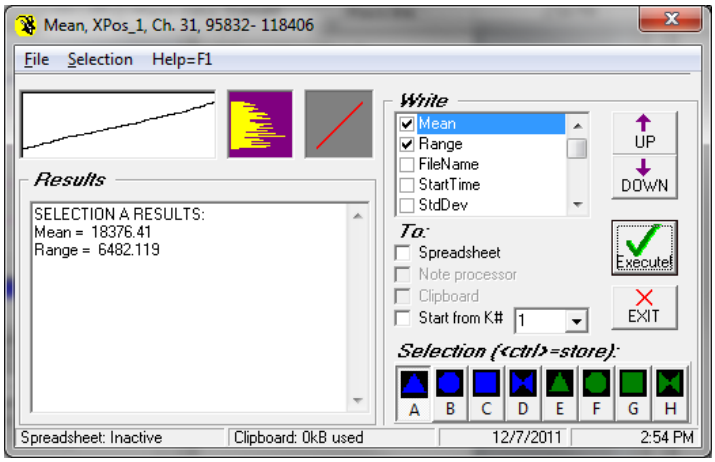
Where in this case the slope has the units beambreaks per hour, because hours were specified as the regression's X axis. (The purple window shows the residuals of the regression).

*Ambulatory Distance / Beam breaks*

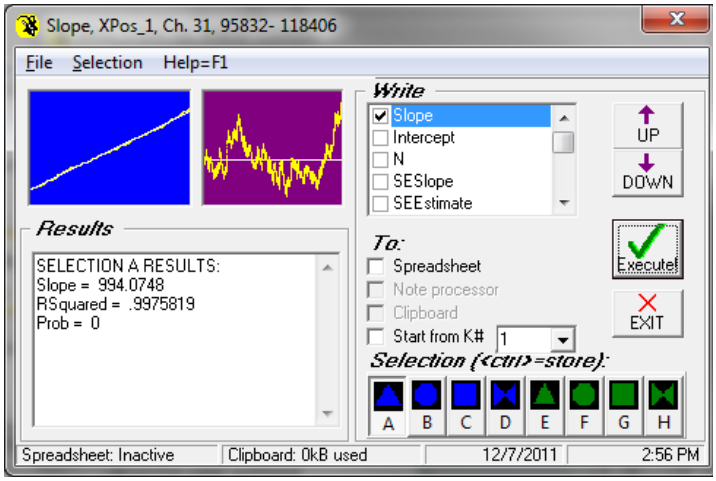
You can easily determine how far a given animal has walked within a cage. In ExpeData, select the region of interest. Make the X or Y axis position active (XPos_1, for example). Then click on TRANSFORM, CONVERSIONS, XY TO DISTANCE. If you want only pedestrian locomotion, set the minimum movement distance (= cutoff speed) to 1 or 2 cm. This is a good idea because even a largely motionless mouse may interrupt a beam from time to time, for example when grooming, causing incremental but significant and misleading increases in apparent ambulatory activity. Specify a destination channel, and the cumulative distance travelled (not including wheel running, of course) will be written to it. The default units are cm, which are **functionally equivalent to beam breaks** because the inter-beam spacing of both X and Y arrays is 1 cm; to obtain meters, divide the transformed channel by 100.

Note that some protocols, such as IMPReSS, may require reporting both the total distance reported (in cm, equivalent to beam breaks) with the minimum movement distance set to 0 cm, and the equivalent distance with the speed cutoff set to, say, 1 or 2 cm. This is to provide two separate but related metrics of activity that separate ambulatory activity from total activity.

Analogously with the above, we see that the animal traveled 6482 cm (64.8 m) during the selected interval.




And we see that its mean locomotion speed (the slope of the cumulative distance line vs. time) is 994 cm/hr or 9.94 m/hr in the selected interval (The purple window shows the residuals of the regression).




## *Position Information*

It is possible to use position information for a variety of purposes, such as determining the proportion of time spent in certain areas of the cage. If you have a need for this type of analysis, please let us know.

You can get a quick look at position data by selecting an area and then clicking VIEW/XYZ DATA and then clicking DRAW.

# THE CAGE & ITS SENSORS: ACCESS CONTROL

## *Food Access Control*

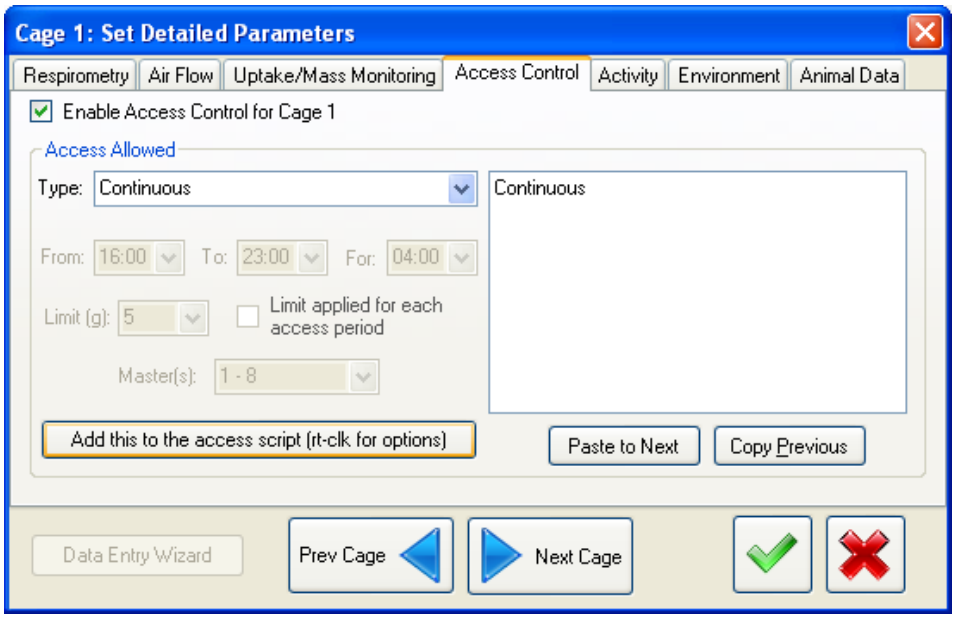
Promethion can implement a wide variety of food access control paradigms. The following list is a snapshot at this stage of the product's development. Other access control strategies will be added based on the input of users and potential users of the system.

Specifying a food access control strategy is simple. Select the type of access control desired, and simply select options from the drop-down boxes that are enabled for each type of access control. When complete, click on ADD. Your access control strategy is translated into an English-language script which is stored for that channel. Many access types allow you to build up arbitrarily complex food access strategies to match your research goals. Each additional access control strategy, such as an access window with a specified food uptake limit, is added to the script when you click on ADD. There is no limit on a script's length.

Presently, there are nine different access strategies that can be selected.

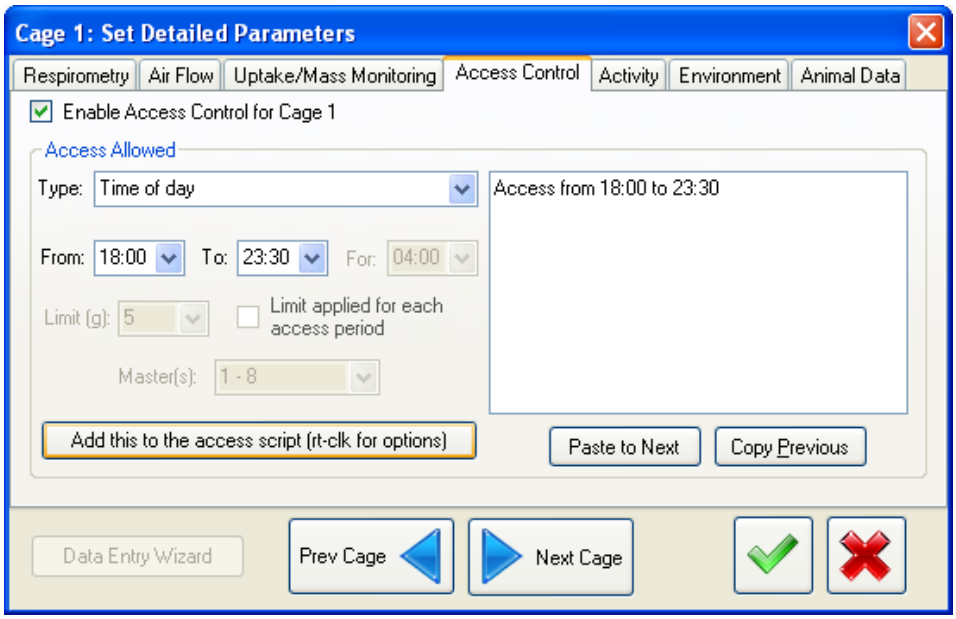
**Continuous access**

This is equivalent to no access control.



**Time of day**

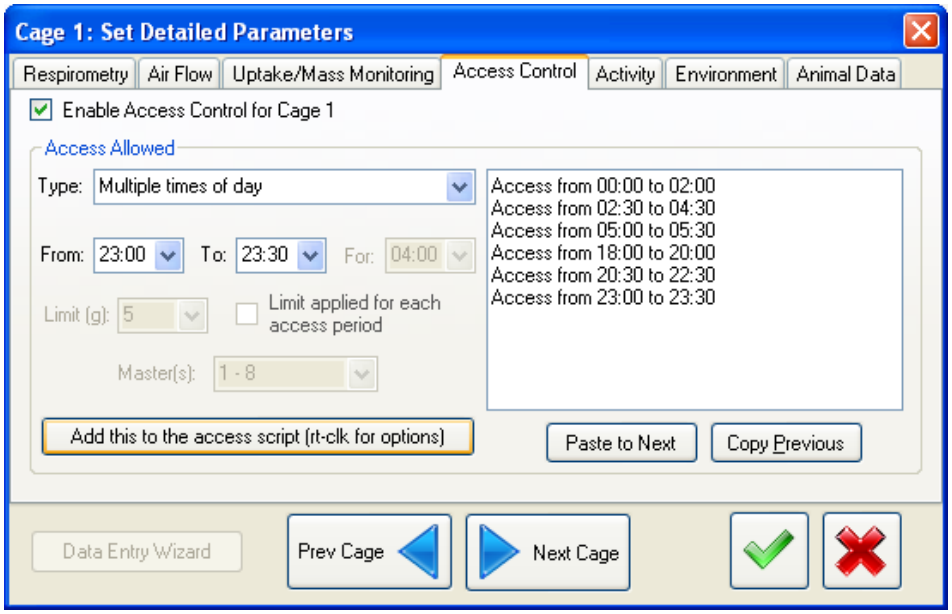
This option allows food access between two specified times. Times can be specified to the nearest minute, as in all subsequent options. The time is taken from the host computer's clock.




If access across midnight is required, set two access periods using the Multiple times of day option. The first access period will be from start time to 11:59PM. The second access period will be from 12:00AM to end time.

**Multiple times of day**

This option allows you to specify separate access times.

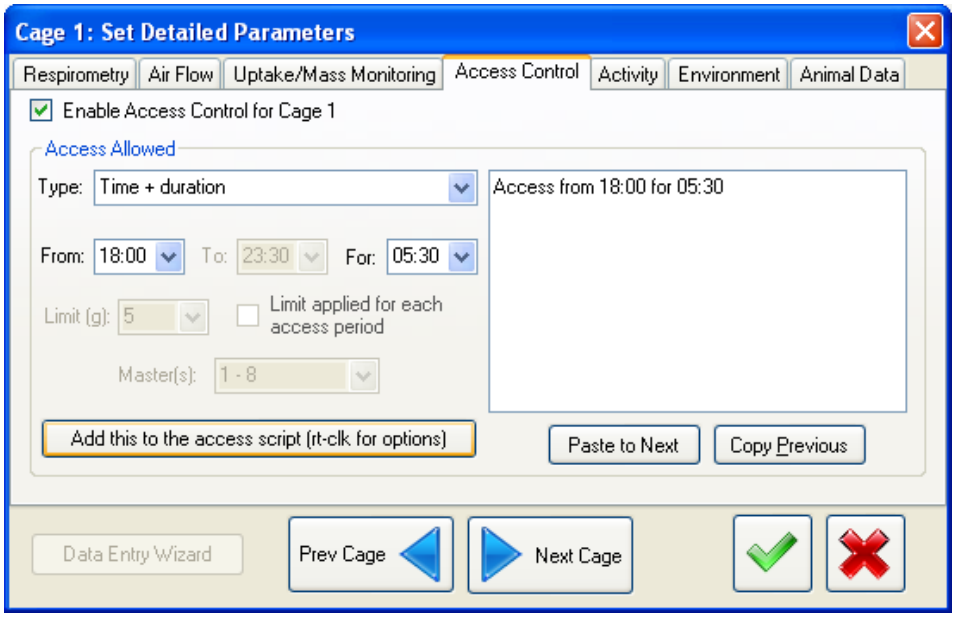


Choose or type in the desired FROM and TO times, and click on ADD to add them to the script. Although you can specify any number of access windows, your access windows are not allowed to overlap. If you specify overlapping access windows, or if any interval overlaps midnight, an error will be flagged and the selection will not proceed.

Note that when the script is complete, you can copy it to other cages (click on COPY TO ALL, PASTE TO NEXT or RT-CLK on the ADD button to get an option menu).

**Time + duration**

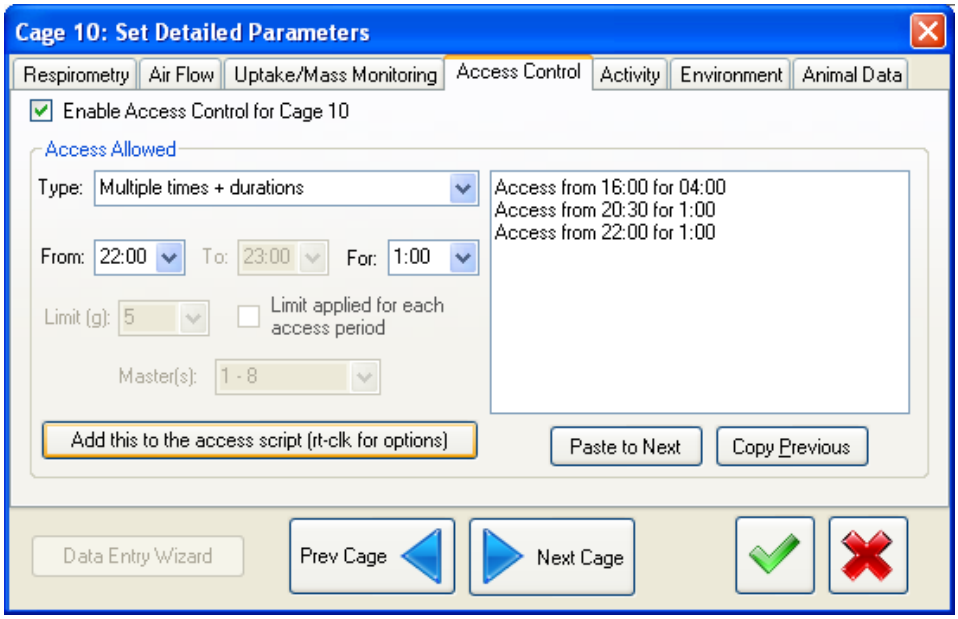
This option allows you to specify an access starting time and an access duration.




The single access window can extend across midnight, if required.

**Multiple times + durations**

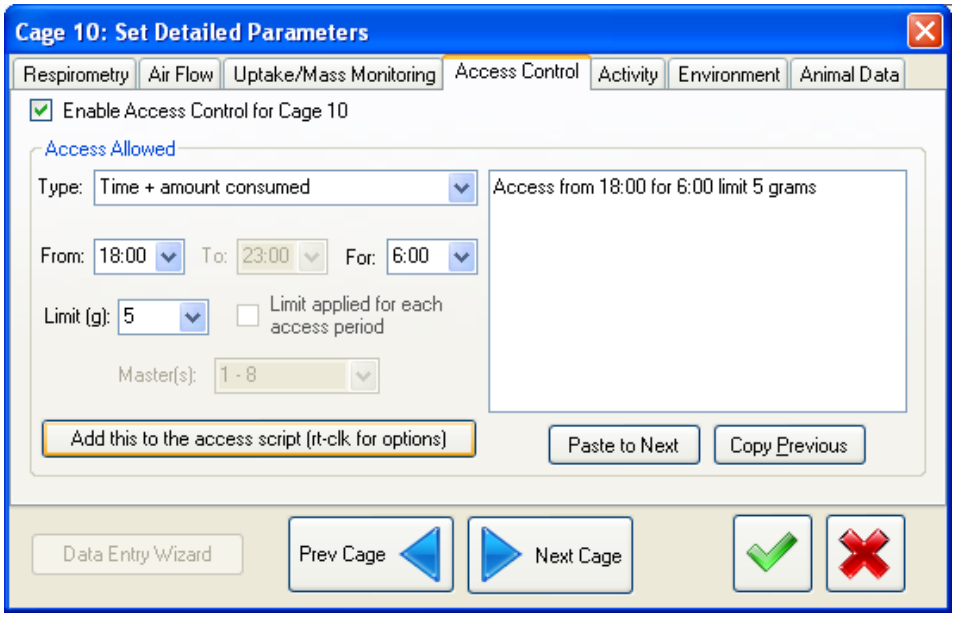
This option allows you to specify any number of separate access times and associated durations.




See notes for “Multiple times of day”, above.

**Time + amount consumed**

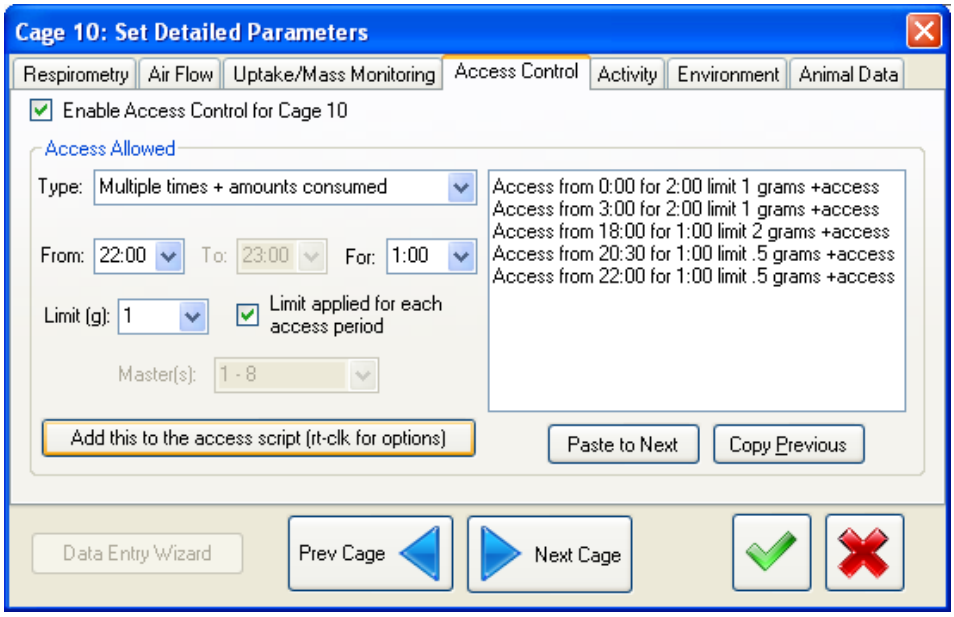
This option allows you to specify both an access window, and a limit to the amount of food that can be consumed within the defined access window



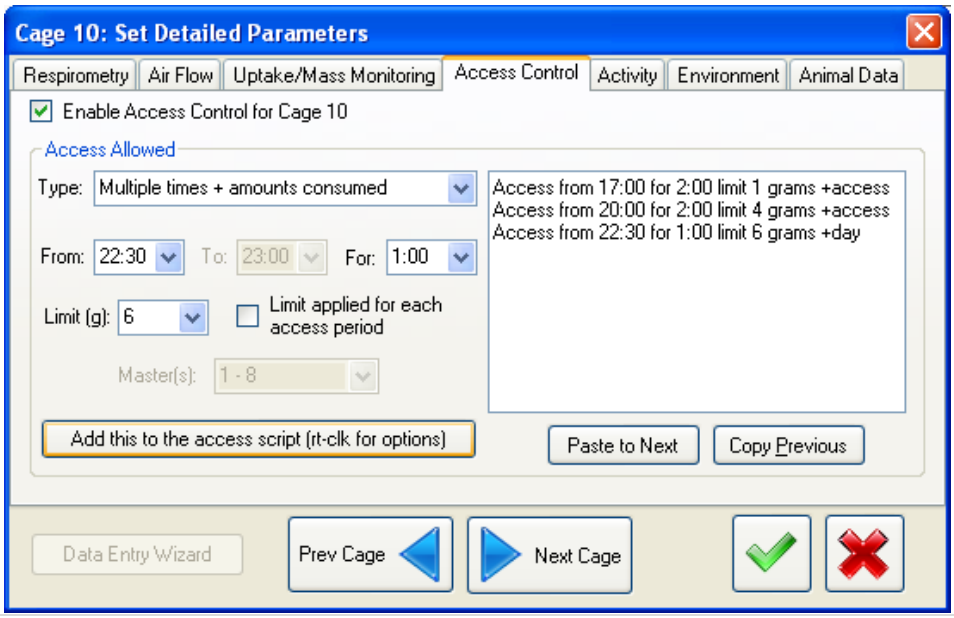
Although the window can straddle midnight, the food limit is calculated on a per day basis. For example, with a time setting of “Access from 18:00 for 12:00”, a mouse may consume 2.8 g of food between midnight and 6 a.m. and then consume 2.2g starting at 6 p.m (18:00), after which food access will be denied.

**Multiple times + amounts consumed**

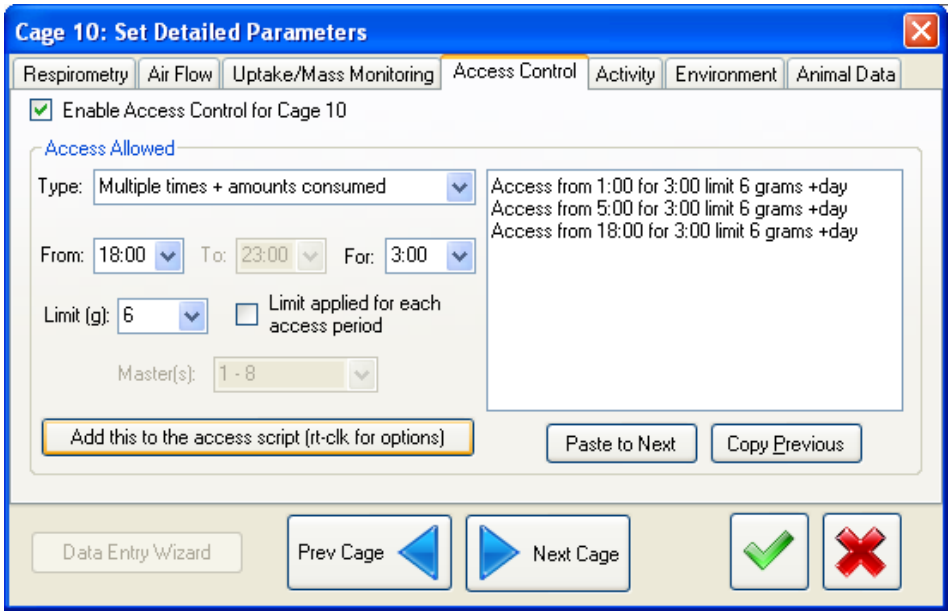
This option allows you to specify any number of separate access times.




Each access window can have an associated food uptake limit, and the food uptake limits can be different between access windows. The food uptake limit can also be specified on a 24-h basis by unchecking the LIMIT APPLIED control for the last access. For example,

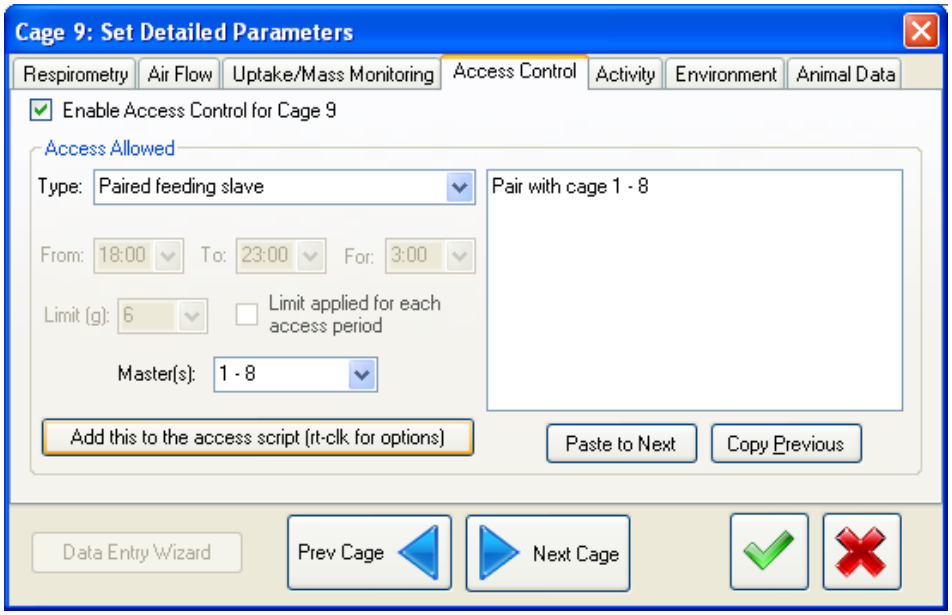


In this case the animal is limited to 1 gram of food in the first access window, 4 grams in the next, and the total eaten across all access windows is limited to 6 grams. Thus, the animal could, for example, eat 1 g and then 2.5 g and then up to another 2.5 g in the three access windows; or 0.5, 4 and 1.5 g, and so on. If window-by-window access limits are not needed, but a 24-h limit is required, simply specify the same uptake limits for all access windows, being careful that the LIMIT APPLIED control is unchecked. For example,



**Paired feeding slave**

You can select any cage or combination of cages as the master for any cage designated as a paired feeding slave. When the mean food uptake of the specified master cages over the previous 24 h is reached, access is removed. The access control scripts of the masters determine the window during which food is accessible by the slave.




The only limitations are that all masters for a given paired feeding slave must share the same food access times or share continuous access, and, of course, that a paired feeding slave cannot have a paired or yoked feeding slave as its master[1]. For the slave, the master selection (selected from the drop-down menu or entered manually in the MASTERS combo box) can be in any of these formats, where cage numbers are used for example only:

| | |
|---|---|
| 1 | [ a single cage] |
| 1-8 | [the mean of cages 1 through 8] |
| -6 | [the mean of cages 1 through 6] |
| 1 3 2 4 6 | [the means of cages 1, 2, 3, 4 and 6; numbers can be in any order] |
| 1-4, 9-12 | [the means of cages 1 through 4 and 9 through 12] |
| 1 1 1 2 | [the mean of cages 1 and 2, with cage 1 weighted 3x in the mean calculation] |
| -4,13- | [the means of cages 1 through 4 and 13 through 16] |

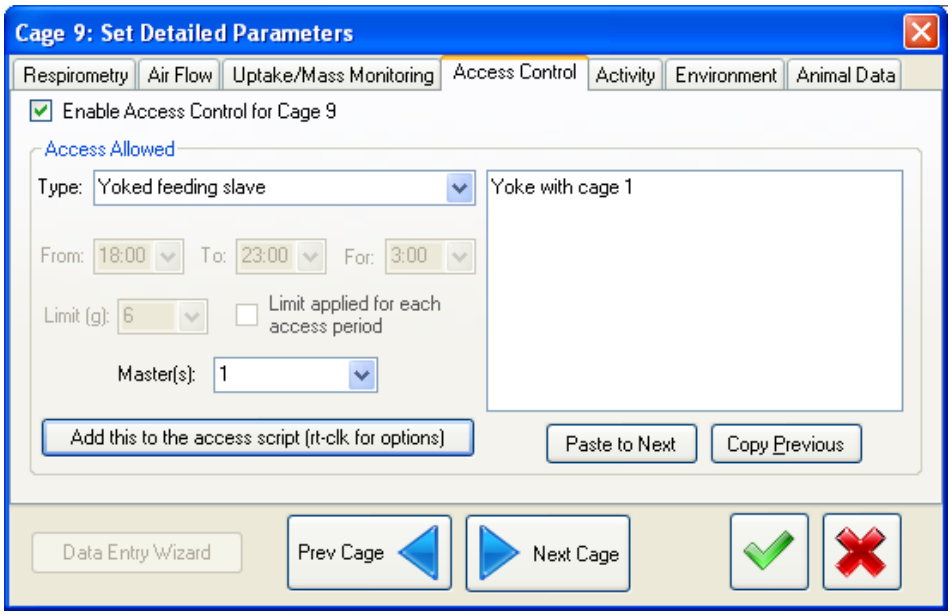
[1] These requirement can be relaxed if required. However, they seem to be sensible precautions.

The masters selection string can be of any length. Spaces must be used to separate numbers so that, for example, 1 and 3 can be distinguished from 13. Extra spaces are ignored. For internal use, dashes ("-") are expanded to yield the intervening cage numbers but these are not displayed in the script. When a dash is used with only one number, the system interprets this as the second number being the maximum cage number (if dash is after number) or the minimum cage number (if dash is before number).

When you select a paired feeding slave, its masters are checked to ensure that they all share the same access script. Therefore, always specify the access types etc. of the masters before specifying the slaves, and make sure that all of the masters for a given slave share identical access scripts. If differences are detected between the masters' scripts, or if a slave of any kind is selected as a paired feeding master, an error will be flagged and the selection will not proceed.

**Yoked feeding slave**

In yoked feeding, variations in food hopper mass in the master cage, which are caused by the master animal's feeding activity, are detected. If these variations persist for > 1 minute, the food access door is raised in the slave animal's cage.




After the master's feeding has ceased for 1 minute, access to food is again removed from the slave cage. Only one cage can act as a master to a given yoked slave. Any cage can, however, act as a master to a yoked slave, including a paired slave or another yoked slave.

**IMPORTANT NOTE:** Before using feeding access controls, ensure that the access doors have been properly adjusted to open and close fully. Detailed instructions for adjusting the AC-1 doors are provided in another

manual.. If access control is attempted before properly adjusting the access doors, it is possible that access control will not function as expected.

# ADVANCED DATA ANALYSIS: INTRODUCTION TO MACROS

## *Don't Worry, Be Happy*

Macros, or analysis scripts, can be a complex subject. Information on macros is included for those who want (or need) to understand how ExpeData's macro language works. If you have well-defined needs that are met by a macro that you currently use, or if you want to use Sable Systems' macro generation services to write macros for you, then you will not need to read through this section. It will nevertheless not actually harm you to know what the macro language is doing, and may even be beneficial.

## If you want to jump straight into executing a macro to extract data from a file, please jump back to Chapter 10.

## *Macro Principles*

A macro can just as easily be called a script. It is simply a series of commands that the macro utility reads, interprets and executes in sequential order. When you invoke the macro utility (TOOLS/MACRO), and load a macro collection, click on VIEW in the macro window to choose your macro, and play it back by clicking on the PLAYBACK icon. The macro is played back, interpreted, and executed line by line. Typical macro actions include removing drift, carrying out mathematical transformations, selecting areas of data, and writing information to ExpeData's built-in spreadsheet. For very simple macros, this might be all you need to know.

But what about writing macros?

If you are operating ExpeData manually and click the RECORD icon, your commands will be executed as usual, but in addition the macro utility will record everything you do. It will record everything using the ExpeData macro language. You can read and edit the resulting macro by clicking on the notepad icon in the macro utility.

When your macro is working to your complete satisfaction, you can save your macro by clicking MACRO/SAVE MACRO COLLECTION in the macro utility's menu bar. A macro can be of any length.

You can add any number of new macros to your macro collection. Simply click MACRO/ADD MACRO. Select which macro to edit, or play back, or record to using the macro selector, a combo box that contains a list of available macros by number.

Within macros, you can add comments to explain what is happening or document new sections. Any line starting with an apostrophe (') is treated as a comment and is ignored by the macro interpreter.

Any comments that are in the first line of the macro are visible as a pop-up window that appears when you hover your mouse over the macro selector box. The first comments in any macro are also visible in a dialog that appears when you click on view in the macro window. You can double-click on the description of the macro that you want to run, and that macro is automatically selected. You may find this to be a useful way to label the macros in your collection.

Advanced users will treat a basic macro, recorded while the RECORD icon is active, as merely the starting point for a more advanced macro. This is because several advanced features are only available when you hand-edit a macro. These include most conditional tests, and all looping structures.

**Conditional tests** allow code associated with them to be executed if a specific test is performed and found to be true. For example, it is possible to test for the presence of a specific channel name or marker, and to execute a block of code if the test is successful (or else if it is not, depending on how the test is configured).

**Looping structures** allow specific blocks of code to be executed a specific number of times. For example, it is trivial to write a macro that looks for food hopper mass channels numbered 1 to 64, e.g. FoodA_1 through FoodA_64, tests for the presence of each such channel, and then, if that channel exists, transforms it to create a channel of cumulative food uptake in grams.

Conditional and looping structures can be nested to any depth. This means, for example, that you can apply a conditional which, if true, applies another conditional which, if true, executes a looping structure within which another conditional is applied, which, if true, executes another looping structure or, if not true, executes a different looping structure. In short, the ExpeData macro environment is a very powerful and adaptable analytical tool. It is also in constant development, and your suggestions for useful extensions to it are always welcome.

## Developing a Macro

There are many ways of developing a macro.

The easiest is to modify an existing macro to do what you need. To read the text for any macro, select it and then click on the notepad icon in the macro window. Then you can cut and paste it into any text editor. (We recommend that you do not modify the macros in the existing Universal Macro Collection unless for a very specific and minor purpose. Any major edits should be saved under a different name.)

You can add as many macros as you wish to a macro collection by simply clicking MACRO/ADD NEW MACRO for each new macro. Describe the purpose of each macro at the very start of each macro, starting each line with an apostrophe; if you do so, the macro's description will be visible if you click on VIEW in the macro window.

*You can load a macro file into a standard text editor such as Notepad. If you do so, you'll see that each macro in the collection begins with [macro X] where X is the number of the macro. The end of each macro is denoted by [/macro X]. Advanced users can edit macros with any text editor they wish, but must adhere to the above format or the macro collection will not be loaded correctly.*

Or you can write a macro from scratch. Below is the general approach to doing so that is easiest and most useful, in our experience.

1. Start with a blank macro (exit from the macro window then click TOOLS/MACRO to start a new collection, or click MACRO/ADD NEW MACRO in an existing macro collection).

2. With the RECORD function of the macro dialog active, go ahead and use ExpeData's prompts and menus to perform your analysis. Most of ExpeData's macro commands are quite complex and it is far easier to have ExpeData write them for you than to write them yourself. For this reason, we are not including exhaustive details on the syntax of each macro command.

3. Do all of the analytical activities you would like to perform on the file as a whole. These typically include calculation of metabolic, uptake or behavior data.

4. If appropriate, for just one cage, do all of the analytical activities you require, including creating interpolated channels (see later), processing uptake channels, and writing to the spreadsheet.

5. Stop recording, and go to the macro editor (click the notebook icon).

6. Edit the per-cage section of your macro to incorporate looping structures to loop through the range of cage numbers you are likely to require. Edit each recorded command within the per-cage section so that the cage number is replaced by the loop variable (for example, if your loop variable is called {cage} and your example code was executed for cage 1, replace each reference to 1 in this section of your recorded macro with {cage}). See the chapter on loop structures.

7. Add a conditional within the loop so that your code is only allowed to execute if the channel it is trying to use actually exists (e.g. if exists_channel_title FoodA_{cage} then…). See the chapter on conditional tests.

8. Test the macro on a freshly reloaded file (hint: if you place the command "reload" [without quotes], at the start of the macro, it will reload the file before doing anything else). Correct as required until it performs to your satisfaction.

The next three chapters describe data transformation, conditional tests and looping structures, and methods for exporting final data.

# ADVANCED DATA ANALYSIS: DATA TRANSFORMATION

## *When can you transform data?*

The typical sequence in data transformation is to make a channel active (view/active) and then choose any of Expedata's vast array of available transformations. These are available under the transform menu on the main Expedata window. **Note that a data channel must be present in memory in order to be transformed.** When Expedata is operating in manual mode, this is assured because the channel must exist in order for it to be made active. When a macro is being run, however, trying to make a channel that does not exist active will result in an error. That is why well-written macros use conditional tests to make sure that a channel exists prior to transforming it. Conditional tests are described in the next chapter.

## *Available transformations*

The available transformations are exhaustively documented in Expedata's on-line help. The online help is available by pressing F1 within Expedata. Help is context sensitive, so don't hesitate to press F1 while operating Expedata in manual mode if you need help.

### Common Transformations

Common transformations include adding a number to a channel, subtracting a number from a channel, dividing a channel by a number - you get the picture. Click on transform in the main Expedata menu to see the range available. Press F1 to obtain a complete list. Remember that all common transformations are executed on the active channel. You can choose the active channel by clicking on view/active.

If you need to know the macro syntax for any transformation or operation, the best way to learn it is to start a blank macro, click record and then perform the operation. Click stop, and then click the notepad icon to see the syntax. Or, easier still, simply look at the text of the macros that came with your system. The macro language is quite simple and intuitive, especially if you have ever done any programming.

### General Transformation

The general transformation is the most powerful transformation available within Expedata. Using the general transformation, you can transform the contents of any channel in memory, including those that are not currently active. In that transformation, you can refer to the contents of any other channel in memory, as well as to numerical values that can be pulled from the remarks field of the file, such as animal mass, and to special variables called user constants that can be assigned values in many different ways. A general transformation can also incorporate a conditional test, and even create a new channel.

Here are some examples. In these examples, we are only showing the text of the transformation itself. Within the macro language, each of these transformations would be preceded by

```
transform general all_samples
```

```
CO2_A = CO2_A/100
```

Convert CO2_A, the CO2 percentage measured by the first CO2 analyzer in a multiplexed system (handling cages 1 – 8), from percentage to fractional concentration

```
CO2_A = CO2_A * BP_A / (BP_A – WVP_A)
```

Convert CO2_A to eliminate the effect of dilution by water vapor (BP_A is barometric pressure, and WVP_A is the partial pressure of water vapor).

```
VO2_A = FR_A * (O2_A – 0.2094 * CO2_A) / (1 – 0.2094)
```

Create a new channel, called VO2_A, from flow rate, delta O2 (O2_A), and delta CO2 (CO2_A). Note that prior to this equation, both VO2_A and VCO2_A were converted from percent absolute concentrations to differential fractional concentrations after correction for water vapor dilution, atmospheric pressure, et cetera. See Lighton (2008) for details.

## Corrections

This dialog contains a host of useful corrections that allow you to smooth, lag-correct, span, and drift-correct data.

## Exotic Transformations

This dialog contains specialized transformations for applying the Z correction (a.k.a. response correction or the instantaneous correction), calculating food or water intake from raw mass data, calculating body mass from intermittent mass measurements, et cetera.

# ADVANCED DATA ANALYSIS: CONDITIONALS AND LOOPING

## *Conditionals*

Conditionals test whether a certain statement is true or not. Conditions are often referred to as Boolean tests. Conditionals cannot be entered when using ExpeData in manual mode. They can only be entered by editing a macro in a text editor. The general syntax for conditionals is in this form:

```
If [test] then
       [executable statements 1]
       Else
       [executable statements 2]
       endif
```

[Test] is a conditional that can be either true or false. If the test is true, the first set of executable statements is executed. If it is false, the optional second set of executable statements is executed. The endif statement signifies the end of the conditional, and must be included.

There are two different types of conditional tests. Unary conditional tests have a statement and a parameter, for example, “exists_channel_name MyChannel” will test for the presence of a channel called MyChannel, and return true if that channel is found, and false if it is not. Binary conditional tests compare two values, for example, “(k1 > 5)” will test whether a user constant (which you will learn about later) has a value greater than five. If K1 is less than or equal to five, the conditional test will return false, otherwise it will return true.

Here are the most commonly used conditionals in the Expedata macro language.

`Exists_Channel_name`
This tests whether a specified channel name exists in the list of channels within the recording that is loaded into Expedata. Example:

```
If Exists_Channel_name FoodA_1 then
       ‘ convert grams to kg
       transform general all_samples expression FoodA_1=FoodA_1/1000
       Else
       ‘ do something else; “else” is optional
       Endif
```

`Exists_Marker`
This tests whether a specified marker exists in the recording that is loaded into Expedata. Example:

```
If Exists_Marker D then
       ‘ implement a calibration calculation in which the
       ‘ marker D occurs; otherwise skip the calculation
       Endif
```

## *Looping*

in general, most advanced data analysis of data extraction procedures require executing a block of commands on the successive series of channels. In Promethion, many of the channels that are recorded contain an embedded reference to the cage to which they refer. Examples include food and water intake, body mass, wheel revolutions and so on. The syntax for forming an embedded reference is very simple. First, the name of the variable or the sensor housed in that channel is given, for example FoodA, water, body mass, VO2, VCO2, or RQ. Then, the cage number is denoted by an_character followed by the cage number, for example VO2_1.

Consider an example in which you are dealing with sixteen cages, each of which may have multiple parameters. Obviously, you could repeat the same block of commands sixteen times, updating the cage number with each repetition. However, that would be extremely inefficient, and if you made any change to the calculations, you would have to repeat that change sixteen times.

There is a better way. By using looping, often in combination with conditionals, you can breeze through the situations very quickly.

The primary loop construct in the macro language is the for-next loop. Here is the syntax:

```
For {cage} = 1 to 16
       [executable statements]
Next {cage}
```

notice the word, in this case cage, enclosed within curly brackets. When the loop is executed, every occurrence of that word will be replaced by a number, in this case from 1 to 16. For example,

```
For {cage} = 1 to 16
Prompt I'm looking at data for cage {cage}!
Next {cage}
```

Would cause the prompt to cycle from cage one through cage sixteen.

Often, to make allowance for the fact that not all channels may always be present, the-for next loop is combined with a conditional. With the conditional in place, the executable statements are only executed if the channels to which they refer actually exist.

```
For {cage} = 1 to 16
If exists_channel_name then
       [executable statements 1]
       Else
       [executable statements 2]
       endif
next {cage}
```

If you look at the text of the macros, you will find dozens of examples of this type of construct. For-next loops can be nested to any level, and you can use any word you wish between the curly brackets. However, be sure to terminate each loop with a next statement followed by the name of the word to which it refers.

Loops are quite versatile. You are not restricted to counting with integers. You can count with floating-point numbers, such as 0.01, and you can use any step size you wish as you proceed through the loop. The default step size is one. To specify different step size, simply follow the “to” variable with the word step followed by the step size, for example:

```
For {cage} = 1 to 16 step 2
Prompt I’m looking at data for cage {cage}!
Next {cage}
```

Using floating-point numbers and small step sizes can be valuable when investigating the optimal settings for food uptake variables, body mass measurement parameters, and so on.

You can also specify an alphabetical variable. To do so, simply use a character instead of a number, for example:

```
For O2_{n} = A to B
Prompt I’m looking at data for channel O2_{n}!
Next {n}
```

# ADVANCED DATA ANALYSIS: EXPORTING FINAL DATA

## *To the ExpeData Spreadsheet*

After loading the universal macro collection, click on VIEW in the macro window. A list of macro descriptions will appear. Choose the macro you wish to run by double-clicking on its description. Most of the macros write directly to the Expedata spreadsheet. Within that spreadsheet, you can do a variety of useful transformations and also use the built-in statistical package.

The statistical package includes a Boolean evaluator, which allows you to select subsets from your data for analysis. These might include, for example, selection by circadian cycle or selection by experimental animal. Data sets can be compared by Student's *t*-test, the Mann-Whitney U test, and by analysis of variance. The statistical package also supports analysis of covariance (ANCOVA). The statistical package is fully prompted on-screen and designed to be easy to use.

Of course, you will need to save your exported data from the spreadsheet into a file. If you choose a comma as the delimiter in the spreadsheet, which is an option at the bottom of the spreadsheet window, you can save your spreadsheet files as comma separated values, or.CSV, files. To do so, simply click on file, save as, and be sure to add the.CSV suffix to your filename. That CSV files can be loaded directly into conventional spreadsheet programs such as Excel.

## *Directly to Excel*

Some of the macros in the universal macro collection are designed to write directly to Excel. They accomplish this by first writing an image of the data to be transferred to Excel into the clipboard, and then pasting the clipboard into Excel and accomplishing miscellaneous formatting tasks under direct program control.

This approach is especially useful for large data files. For example, if you want to export every metabolic reading from a multiplexed recording that is both long and that spans many days in duration, you will quickly reach the limits of Expedata is built-in spreadsheet. Microsoft Excel, practically speaking, has no such limitations.

Of course, using this approach does require that you have a copy of Excel on your computer. However, in the event that you do not have a copy on your computer, the macro will complain but an image of all of the exported data will still be on the spreadsheet. From there, you can paste it into any program you wish.