Table of Contents

1   Introduction: what is MagStat?

MagStat is a newly developed application written in JavaFX which makes use of JAPC, a unified Java API for Parameter Control relating to equipment access. JAPC was developed at CERN and then adapted at GSI. MagStat allows the user to perform an online monitoring of the status of magnets (from now on often also called as "devices") belonging to a selected beamline from the ion source up to the experiment.

2   How to start the application

In order to start MagStat, open a new terminal and perform the following instructions:

- Type "`scheidenb@asl740.acc.gsi.de`" followed by password to connect to an "asl" machine.

- Navigate to "`/common/usr/cscoap/bin`".

- Start the launcher typing "`./launcher pro`".

- Select the tab "FRS".

- Select "FRS MagStat" from the menu to start the application.

3   The working principle

Figure 1 depicts a simple schematic describing the working principle of MagStat. Data treated by the application come from the following sources:

- LSA (LHC Software Architecture) database, related to logical devices.

- FESA (Front End Software Architecture) database, related to hardware devices.

MagStat executes a sequence of tasks which can be summarized as follows:

- *MagnetDataProcessorLSA.class* retrieves, from the LSA database, a list of devices belonging to a certain beamline chosen via the Pattern and Chain selectors (see Section 4.1). For each device, a set of parameters (electric current set value I_{soll}, integrated magnetic field BL_{soll} and integrated deflection angle KL_{soll}) are loaded. The value of the parameters is theoretical and dependent on the optics model adopted. Parameters are updated every 10 seconds by the *DeviceUpdateLSA.class*. This means that any change in the value of the parameters is displayed only once the update cycle has been completed.

- *MagnetDataProcessorFESA.class* retrieves, from the FESA database, a list of devices belonging to a certain beamline chosen via the Pattern and Chain selectors (see Section 4.1). For each device, a set of actual parameters (see Table 1) is loaded.

- *MagnetDataJoiner.class* merges, for each device, the parameters loaded from the LSA and FESA database. The values are passed to the *MagStatMainController.class* and then displayed in the table as shown in Figure 2.

- *DeviceUpdateFESA.class* starts an online monitoring of the FESA parameters values. In this case, the update is performed whenever a change in any of the parameters value is detected.

4   The main user interface and its components

Figure 2 shows the MagStat main user interface. It is divided into four regions each of which contains the following items:

- Top: pattern and chain selectors and update DCValue check box;

- Center-Left: the table for online monitoring of the devices status;

- Right: action buttons;

- Bottom: status bar for messaging including a data saving LED indicator.
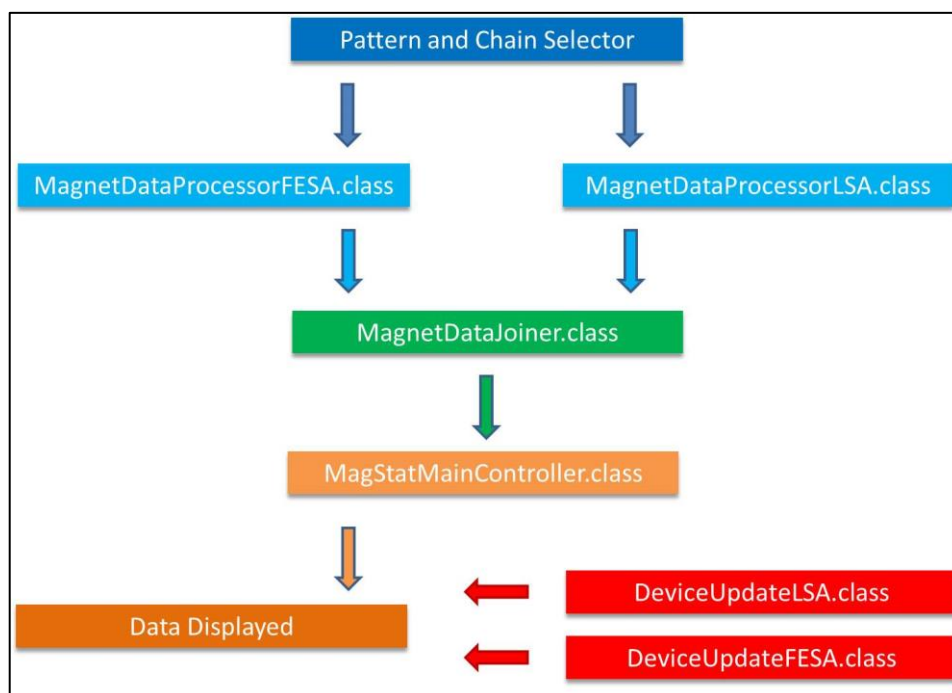

Figure 1: schematic of the MagStat working principle.

Table 1: the list of parameters retrieved from the FESA database.

Parameter	Description
Acquisition/current	It describes the value of the actual electric current I_{ist} read out from the magnet power supply when DCMode = OFF.
Acquisition/voltage	It represents the value of the actual voltage read out from the magnet Hall Probe (if provided) when DCMode = OFF.
DCMode/mode	It assumes values *true* and *false* for DCMode = ON/OFF respectively. When DCMode = ON it allows the magnet to be precycled or calibrated.
Status/status	It shows the actual status of the magnet.
Status/detailedStatus	It shows the detailed status of the magnet in terms of boolean values (not displayed at the moment in the current version of MagStat).
Status/detailedStatus_labels	It shows a description of the detailed status of the magnet in terms of string values (not displayed at the moment in the current version of MagStat).
DCValueAcq/current	It represents the value of the electric current read out from the magnet power supply when the DCMode = ON.
DCValueAcq/voltage	It represents the value of the voltage read out from the magnet power supply when the DCMode = ON (not yet available in the current version of MagStat).

4.1 Pattern and Chain selectors

Pattern and Chain selectors contain a list of available patterns (both active and inactive) and chains. A pattern represents a group of beam production chains that are executed periodically in a certain order (so far each pattern has only one beam production chain associated). A beam production chain defines a sequence and parameters of beamlines.

A beamline is selected by choosing a combination of pattern and chain. If devices related to the selected pattern and chin are available, their associated parameters (see Section 3) will be loaded and the relative values displayed in the table as shown in Figure 2. If the loading process terminates with no errors, a message in the status bar will indicate the total number of loaded devices. Only in case of active patterns, data displayed will be constantly updated.

Figure 2: the main user interface.

4.2 Update DCValue check box

By ticking the update DCValue check box, it will be possible to track changes of the electric current value of a power supply while performing a calibration or a precycling procedure of a certain magnet. The aforementioned procedures can be executed only if the DCMode values of the power supplies relative to the magnets of interest are set to true. In the current version, electric current updates are performed every 2 seconds. In order to minimize the number of processes within MagStat, it is highly recommended to untick the check box once the procedure of calibration or precycling is completed.

4.3 Table

The main table consists of eleven columns each of which containing the following information:

- **Device ID**: it represents the unique name which identifies a device.

- **Status**: it shows the current status of the device. It assumes the following values:
  - 0: Unknown.
  - 1: Ok.
  - 2: Warning.
  - 3: Error.

- **DC_Mode**: it reports whether the device DCMode is ON ("true") or OFF ("false").

- **I_ist [DC]**: it shows the electric current values acquired from the power supply when DC_Mode of the device is set to true.

- **I_ist**: it is the value of the electric current acquired from the power supply when the DC_Mode is false.

- **I_soll**: it represents the theoretical value of the electric current imported from the model.

- **ΔI/I**: it reports the difference of the electric current values $I_{ist} - I_{soll}$ divided by I_{ist}.

- **Bρ**: it shows the rigidity obtained dividing the actual integrated magnetic field B_{ist} by the integrated deflection angle KL_{soll}. B_{ist} is calculated back from I_{ist}.

- **BL_soll**: it reports the theoretical value of the integrated magnetic field imported from the model.

- **KL_soll**: it shows the theoretical value of the integrated deflection angle of the magnetic field imported from the model.

- **Hall Probe**: it represents the value of the integrated magnetic field retrieved back from the voltage value read out by the Hall Probe. Devices equipped with hall probe are at the moment: GTS3MU1, GTS3MU2, GTS4MU1, GHFSMU1, GTS5MU1, GTS6MU1 and GTS7MU1.

The table shows data sorted by device location from the synchrotron SIS down to the experimental area of interest. More details about the flow of data and how they are calculated are given in Section 5.

4.4 Action Buttons

4.4.1 Refresh

The Refresh button updates the list of the Pattern and Chain selectors. Once the list is ready, data related to the first pattern and chain will be automatically loaded and displayed in the table. The user will be then able to switch to another pattern and chain if needed.

4.4.2 Save Data

The Save Data dialog is shown in Figure 3. Two are the modalities by which data can be saved on disk:

Figure 3: the Save Data dialog.

- **Manual**: the user is allowed to edit the default path and the file name. The former is initially set to `/home/rifr/scheidenb/lnx/magstat`. Use of special characters is prevented for both fields. The button **Save** will be enabled only once the two fields are filled in. Such a modality allows saving data just once. Each saved file will follow a pattern name like `<filename>_<date>_<time>` where the system date and time will be automatically added during the data saving.

- **Auto**: the user is allowed to change the default directory and the frequency at which files should be saved. The default value is 60 seconds although it is possible to decrease it to up to one file saved per second. The **Start** button allows starting the auto saving modality and it will be enabled only once the file path and frequency fields are filled in. The **Stop** button allows the interruption of a previously activated auto saving modality. In this case file will follow a pattern name like `MagStat_file_<date>_<time>` where the system date and time will be automatically added during the data saving.

In case the chosen file path doesn't exist a message error will be prompted. By closing it, the Save Data dialog will continue to remain open. The **Close** button in the Save Dialog allows the user to return to the main user interface.

Figure 4 shows the structure of a saved file. The first line reports the selected pattern and chain while the second one the name of the optics adopted. Next it follows the header with the data acquired from the MagStat main table. Data corresponding to parameters not available are replaced with a dummy value of "999.000000".

4.4.3 Print

Figure 5 displays the Print dialog. The function is enabled only if the main table contains a list of devices for a certain selected pattern and chain. Regardless of the number of listed devices, data will be automatically fitted in one A4 page. The Printer selector allows the user to choose one among the printers available within the network. By default, "Printer p293" located in the Messhütte is selected. The **Print** button prints out the data currently displayed in the table while the **Close** button let the user to return to the main user interface.

Figure 4: the structure of a saved file.

Figure 5: the Print dialog.

4.4.4 Graph

MagStat is able to provide a graphical monitoring of the parameter *ΔI/I* for each device belonging to the selected pattern and chain. As reported in Figure 6, the Graph dialog is divided in three main areas. The upper one displays the currently selected pattern and chain. The middle one contains the chart with data. The bottom area contains input fields and buttons with related functionalities.

Inside the chart, each magnet type is identified with a different color as follows:

- **blue**: steerers;
- **green**: dipoles;
- **yellow**: quadrupoles;
- **red**: sextupoles;

The x axis shows the name of the magnets. Regardless the number of devices, names will be automatically fitted to the x axis length to make all of them simultaneously visible. The y axis reports the *ΔI/I* ratio values for each device. To prevent the ratio from being too large (in case of I_{ist} value close to zero) or equal to infinity (if data are unavailable), a dummy value of ±100 will be assigned to the ratio.
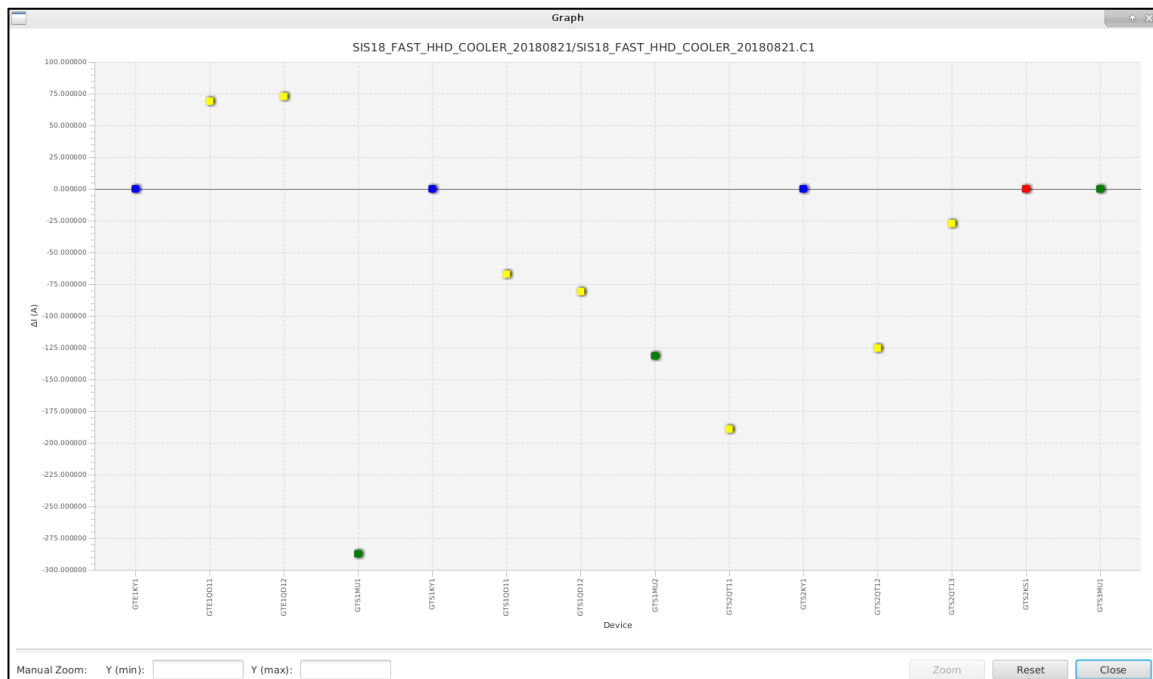

Figure 6: the Graph dialog [a more recent picture is needed].

The input fields Y (min) and Y (max) allow the user to enter manually a new y axis range. The range can be also changed automatically via mouse wheel at a fixed scaling factor of 2. The lower and upper limits and tick numbers of the y axis are calculated by using a developed algorithm.

The Graph dialog contains the following action buttons:

- **Zoom**: it performs the adjustment of the y axis according to the upper and lower limits entered in the Y (min) and Y (max) input fields. The functionality is enabled only at the condition of Y (min) > Y (max).

- **Reset**: it restores the y axis to the initial default values.

- **Close**: it closes the Graph dialog and returns to the main user interface.

4.4.5 About

As reported in Figure 7, the About dialog provides general information about MagStat (Version, Authors, Contacts).

4.4.6 Exit

MagStat and all its related running processes will be closed.

4.5 Status bar

The status bar displays messages to the user whenever a certain event occurs. It is provided of a red LED located to the right side which switches on to indicate that data are currently being saved on disk with auto saving modality (see Section 4.4.2).

Figure 7: the About dialog.

5   The flow of data

Table 2 reports the polynomial functions related to different devices (dipole type) GTS3MU1, GTS3MU2, GTS4MU1 and GHFSMU1. For each specific range, these polynomials represent calibration functions as:

$$I = \mathrm{f}(BL) \tag{1}$$

where I represents the electric current read out by the device power supply and BL the related integrated magnetic field.

From the knowledge of the rigidity $B\rho$ previously calculated by simulations, it is possible to obtain the integrated field BL_{soll} related to a specific device. This can be accomplished multiplying the rigidity $B\rho$ by the integrated deflection angle KL_{soll} whose values are given by the adopted model. In case of dipoles, KL_{soll} assumes the following expression:

$$KL_{soll} = KL_{theo} + HKick \tag{2}$$

where $HKick$ represents a correction factor which is added to the theoretical value KL_{theo}.

Eq. (1) can now be used to calculate the electric current I_{soll} that should circulate in the dipole in order to achieve the BL_{soll} value as previously estimated. Let us consider for the sake of simplicity a value of BL_{soll} equal to 0.2 Tm for the dipole GTS3MU1. As Table 2 shows, such a value falls within the first range and therefore the polynomial function which has to be used for this case is p(x) = 83.41466x. Assuming x as the parameter BL_{soll}, the polynomial function provides a value of electric current I_{soll} equal to 16.7 A. This value will be now sent to the dipole power supply, set, and then read out and displayed in the table as actual electric current I_{ist}. Ideally, I_{ist} and I_{soll} should be the same although little differences are normally observed. In case of larger discrepancies, these may only depend on the power supply of the device.

Knowing I_{ist} it is possible to calculate the actual integrated magnetic field BL_{ist} by finding the root within the domain of the polynomial function. This can be achieved by a numerical algorithm which makes use of the so called *secant method*. The implemented algorithm provides the root value with a precision of $5 \cdot 10^{-8}$ over a maximum number of iterations of 1000. The calculated BL_{ist} divided by KL_{soll} provides the actual rigidity $B\rho$ whose values are reported in the main table.

Polynomial functions are also used to convert the Hall Probe voltage in units of [Tm]. In this case, the secant method has to be applied twice. At first, the voltage, as given by the Hall Probe, is converted into an electric current. Next, the electric current is converted into integrated magnetic field.

Table 2: list of polynomial functions for different dipoles and range of integrated magnetic field.

GTS3MU1

[0.0, 0.359649] p(x) = 83.41466x

[0.359649, 1.14358] p(x) = 29.99999900931659 + 91.55920042103871x - 4.503703087043999x^2 + 1.950548x^3

[1.14358, 7.98013] p(x) = 99.94806554856709 + 88.09413874875294x - 0.3048247341778x^2 + 0.03758503x^3

[7.98013, 9.90935] p(x) = 699.9704103470119 + 89.19624406317514x + 6.012508737836997x^2 + 0.7703482999999999x^3]

GTS3MU2

[0.0, 0.362794] p(x) = 82.69224x

[0.362794, 2.28816] p(x) = 30.0002502199328 + 89.50746064573985x - 1.1152554912048x^2 + 0.2474136x^3

[2.28816, 7.39822] p(x) = 199.9664743119181 + 87.96442413744266x - 0.26900132452159997x^2 + 0.05739658x^3

[7.39822, 9.78032] p(x) = 650.1042925370152 + 89.71151664104215x + 2.753068733620001x^2 + 1.5143569999999997x^3

GTS4MU1

[0.0, 0.351703] p(x) = 85.29924x

[0.351703, 1.13717] p(x) = 29.999999427119146 + 91.34310094890546x - 4.823123922419001x^2 + 2.4824089999999996x^3

[1.13717, 6.826] p(x) = 99.97430082042935 + 88.36092107938909x - 0.297300373864x^2 + 0.0389736x^3

[6.826, 9.7613] p(x) = 600.1982663316309 + 88.76218301664406x - 0.5173424059999974x^2 + 1.724723x^3

GHFSMU1

[0.0, 0.359283] p(x) = 83.49964x,

[0.359283, 1.14712] p(x) = 30.00000015617284 + 90.91272504080229x - 4.851521225952x^2 + 2.709152x^3

[1.14712, 7.40451] p(x) = 99.93790590543178 + 88.31291447718758x - 0.3520025595895999x^2 + 0.04616088999999999x^3

[7.40451, 9.78226] p(x) = 650.0736285797697 + 89.33003538942677x + 2.641170357609994x^2 + 1.676937x^3