

# GCN-клиент для телескопов САО.

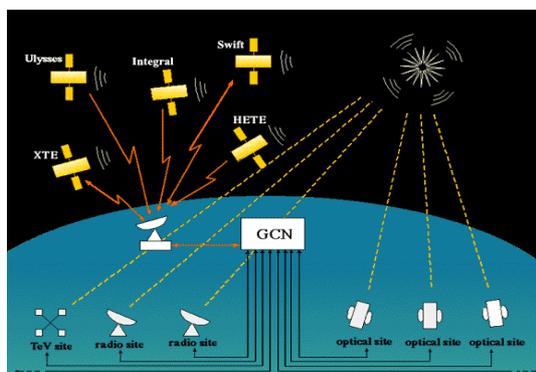
[В.Шергин](#)

2014-2019г.

Для оперативного наведения телескопов по координатам GRB-событий, разработано клиентское программное обеспечение для работы с системой серверов GCN/TAN. Разработка выполнена на языке *Python* с использованием пакета [pygcn](#).

## О системе GCN.

Система [GCN/TAN](#) (*GRB Coordinates Network / Transient Astronomy Network*) разработана и функционирует в [NASA's Goddard Space Flight Center](#).



Главная функция системы это сеть по которой в режиме близком к реальному времени (т.е. за секунды или минуты) распространяются сообщения о GRB (или иных транзиентах) обнаруженных различными космическими аппаратами (Swift, Fermi, MAXI, INTEGRAL, IPN и т.д.) или наземными средствами.

Сообщения распространяются как по e-mail, так и через подключение к TCP-сокету. В последнем случае можно получать либо бинарные пакеты(160 байт) либо VOEvent-ы (XML-тексты).

## О пакете *pygcn* (*anonymous VOEvent client for receiving GCN/TAN notices in XML format*).

Пакет [pygcn](#) написан на языке *Python*.

Автор [Leo Singer](#) (*lsinger*) из California institute of Technology.

Совместим с версиями *Python*  $\geq 2.6$ .

Этот пакет разработан как простой клиент, который анонимно подключается к серверу и прослушивает VOEvent-ы (<http://www.ivoa.net/documents/VOEvent>) т.е. сообщения в XML-формате. Он реализует TCP/IP VOEvent Transport Protocol (<http://www.ivoa.net/documents/Notes/VOEventTransport>).

Для его использования необходимо написать модуль *handler*, которому он будет передавать на обработку каждый полученный VOEvent-пакет.

В настоящее время в системе GCN имеется четыре сервера допускающих анонимное TCP-подключение по протоколу VOEvent:

IP-адрес	положение(сеть)	протокол
209.208.78.170	Atlantic.net	VOEvent v1.1

50.116.49.68	Linode.com	VOEvent v1.1
68.169.57.253	eApps.com	VOEvent v2.0
45.58.43.186	??? (с 24.11.2015)	VOEvent v2.0

Формат XML-сообщений у них немного отличается, что необходимо учитывать при их разборе в модуле *handler*.

Имеется одно существенное ограничение доступа — с одного клиентского IP-адреса сервер принимает только одно TCP-соединение.

Никаких проху-возможностей не предусмотрено, т.е. для обращения из локальной сети САО клиент должен иметь прямой выход в Internet, хотя бы через NAT.

## Описание программного пакета для телескопов САО.

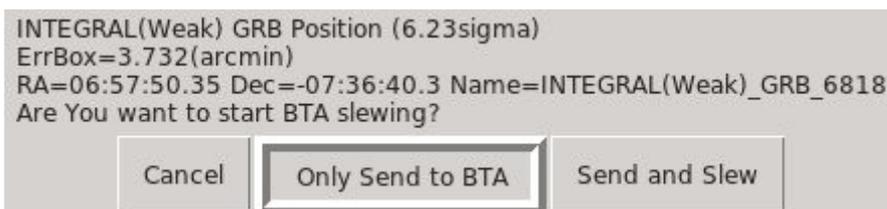
Все программы написаны на языке *Python*. Ставилась задача разработки приложения независимого от типа ОС и при этом использующего только штатные средства основного пакета *Python*. Реальная проверка работоспособности производилась в *Linux Python2.6/2.7* в *Windows Python2.7*. Дополнительно может потребоваться установка пакета *PyAudio*, если у компьютера есть колонки и хочется слышать предупреждения. В ОС *Linux* пакет *Python* ставится по частям, поэтому необходимо убедиться что в системе установлен пакет *python-lxml*, иначе программа может работать неправильно.

Для быстрого наведения телескопов в программе реализована связь с системами управления (TCS) двух инструментов САО — Цейсс-1000 и БТА.

### Принцип работы.

Приняты следующие принципы организации взаимодействия с наблюдателями на телескопах.

- Не предусматривается (по крайней мере пока) никакого автоматического наведения телескопов, т.к. имеется много параметров которые может оценить только опытный наблюдатель (и решить стоит ли прерывать текущие наблюдения). Например: какой источник сообщения, уровень обнаружения GRB, точность координат (ErrBox), текущее положение на небе, погода и т.д.
- После приема от сервера сообщения которое может заинтересовать наблюдателя, на экране появляется диалоговое окно типа:



и наблюдателю предлагается либо использовать эти координаты, либо проигнорировать.

### Источники сообщений.

Рассматриваются сообщения (*notices*) от следующих источников:

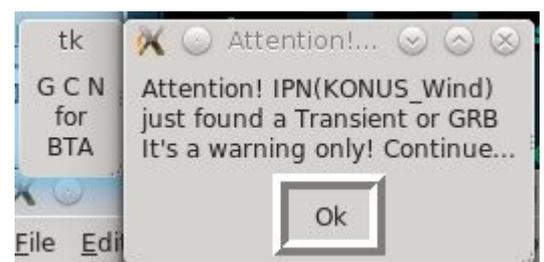
Космический гамма-телескоп **Fermi** (инструменты GBM и LAT).  
 Инструменты SuperAGILE и MCAL спутника **AGILE** (*Astro-rivelatore Gamma a Immagini LEggero*).  
 Спутник **INTEGRAL** (*INTErnational Gamma-Ray Astrophysics Laboratory*).  
 Система **IPN** (*InterPlanetary Network*) объединяющая космические аппараты с гамма-датчиками (например аппарат WIND с инструментом KONUS).  
**MAXI** (*Monitor of All-sky X-ray Image*) на МКС.  
 Орбитальная обсерватория **Swift** (инструменты BAT, XRT и UVOT).  
**CALET** (*CALorimetric Electron Telescope*) на МКС.  
**AMON** (*Astrophysical Multimessenger Observatory Network*) **IceCube** нейтринная обсерватория.  
**SNEWS** (*SuperNova Early Warning System*) система из 7-ми детекторов: Super-K(Япония), LVD(Италия), IceCube(Юж.полюс), KamLAND(Япония), Borexino(Италия), DayaBay(Китай), HALO(Канада).  
**LVC** (*LIGO-Virgo Collaboration*) детекторы гравитационных волн.  
 GRB Counterparts — сообщения о вторичных (*follow-up*) наблюдениях с уточнением координат.

## Фильтрация сообщений.

В сети GCN циркулирует большое количество сообщений. Более 99% из них не представляют интереса для наблюдателей. Это, например, тестовые сообщения (для отладки программ), координаты малозначительных (*sub-sub-threshold*,  $\sigma < 6$ ) пиков, координаты совпадающие с каталогами, сообщения о наведении (*slewing*) спутников и другие служебные сообщения. Поэтому в любой клиентской программе необходима фильтрация сообщений соответствующая ее задачам.

В данном приложении приняты следующие принципы. Все типы сообщений разбиты на группы:

- Предупреждения (*warnings*). Это сообщения о вспышках, либо от детектора без координат, либо от координатного но предварительные (*alerts*) без координат, (например «Swift BAT GRB Alert»), либо точность координат не достаточна для наших телескопов (например ErrBox - градусы, как у Fermi GBM). Наблюдателю выдаются предупреждающие сообщения типа:



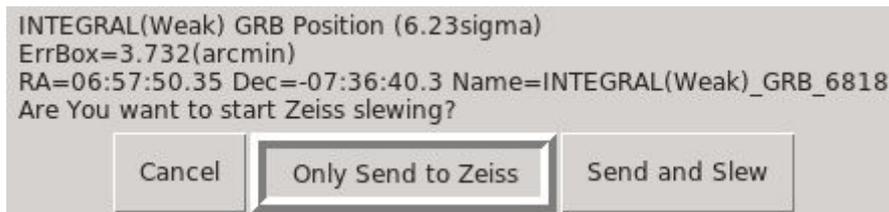
Таймаут ожидания реакции наблюдателя — 10 минут. Смысл в том чтобы наблюдатель был предупрежден о происходящем и о возможном появлении сообщения с хорошими координатами. Кроме того координаты, если они есть, остаются в файле протокола и могут быть использованы в дальнейшем.

- GRB с хорошими координатами ( ErrBox — минуты или секунды). Сюда относим:

INTEGRAL (Wakeup, Weak, Refined, Offline)  
 SuperAGILE (Wakeup, Ground, Refined)  
 Fermi LAT (Ground, Offline)  
 Swift BAT  
 Swift XRT

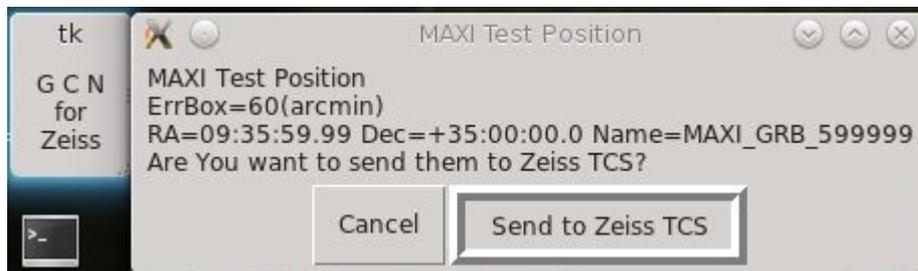
## Swift UVOT GRB Counterpart

Наблюдателю представляется диалоговое окно с предложением отправить координаты системе управления телескопа.



Таймаут ожидания реакции наблюдателя — 1 час. Обычно такие сообщения идут сериями с уточнением координат на этапах обработки сначала на самом аппарате, а затем наземными средствами.

- Тестовые сообщения. Имитация реальных событий от разных аппаратов. Появляются в среднем раз в 2 часа. В данной программе подключаются по желанию для отладки и тренинга.



- Несколько сообщений полезность которых пока не ясна. Информация о них на всякий случай только лишь записывается в файл протокола.
- Самая большая группа сообщений признанных заведомо бесполезными — просто пропускаются.
- Неопознанные программой типы сообщений. XML-текст VOEvent-а записывается в файл на диске для последующего анализа автором программы и включения в одну из групп.

### Запуск программы.

Основная программа называется *gcn\_monitor.py*. Предполагается что она будет вставляться в Автозапуск, стартовать вместе с оболочкой пользователя-наблюдателя и работать непрерывно. При запуске без параметров она покажет справку:

```
GCN/TAN client monitor usage:
gcn_monitor.py [-h] [Tel=...] [Host=...] [User=...] [Pass=...]
[Proxy=...] [Test=Yes] [Audio=Yes]
  Help or -h - show this text
  Tel - telescope name: 'BTA' or 'Zeiss' (default - Zeiss)
  Host - TCS site name to connect to
  User - for Zeiss TCS only (access level>=2 for 'Send'; >=3 for
'Slew')
  Pass - users password for Zeiss TCS or
        access level password for BTA (>=2 for 'Send'; >=4 for
```

```
'Slew')
Proxy - name or address of relay computer where gcn_proxy.py
was started
Test - utilize GCN Test-messages for debugging (default - No)
Audio- play WAV-signal on GCN alert (if PyAudio
installed, default - No)
```

Ее параметры задаются в форме «Имя=Значение»:

**Tel** — выбор варианта на систему какого телескопа настраиваться. Варианты: «BTA» или «Zeiss».

**Host** — сетевое имя или IP-адрес сервера. Для Цейсс-1000 это имя XML-RPC сервера системы управления. Для БТА — имя HTTP сервера на котором расположена программа **bta\_send.cgi** для связи с его системой управления.

**User** — только для Цейсс-1000 у которого доступ к системе управления по имени пользователя. Уровень доступа этого пользователя должен быть не менее 2 для отправки координат и не менее 3 для старта наведения.

**Pass** — для Цейсс-1000 это пароль выбранного пользователя, а для БТА пароль определяющий уровень доступа к системе управления. Он должен быть не менее 2 для отправки координат и не менее 4 для старта наведения.

**Proxy** — если данный компьютер не имеет выхода в Интернет, тут можно задать сетевое имя или IP-адрес компьютера имеющего выход в Интернет, на котором работает специальная программа-ретранслятор **gcn\_proxy.py**.

**Test** — «yes» если нужно воспринимать тестовые GCN-сообщения для отладки или обучения.

**Audio** — «yes» если нужно проигрывать файл *alarm.wav* при выводе диалоговых окон на экран. Это работает только если в *Python* установлен пакет *PyAudio*, иначе игнорируется.

## Состав пакета, файлы.

### **gcn\_monitor.py**

Стартовый модуль приложения. Использует *gcn\_client.py* и *gcn\_alerts.py*. Читает параметры, настраивает и запускает процессы (threads) прослушивания сообщений в сети и управления диалоговыми окнами.

### **gcn\_client.py**

Предоставляет функцию *handler()* для *pygcn*. Использует обращения к *gcn\_alerts.py* для передачи команд в thread управления окнами для привлечения внимания наблюдателя. В этом модуле реализована логика фильтра GCN-сообщений и извлечение данных из XML-текстов.

### **gcn\_alerts.py**

Реализация параллельного процесса (thread) управления диалоговыми окнами на экране. Может проигрывать звук из файла *alarm.wav*. Использует *gcn\_zeiss.py* и *gcn\_bta.py* для доступа к системам управления телескопов.

### **gcn\_zeiss.py**

Модуль выполнения команд через XML-RPC сервер системы управления Цейсс-1000.

### **gcn\_bta.py**

Доступ к системе управления БТА по протоколу HTTP через CGI-программу *bta\_send.cgi*.

### **alarm.wav**

Звуковой файл для проигрывания при появлении сообщения наблюдателю. Можно заменить его на любой свой.

### **gcn\_client\_bta.log** и **gcn\_client\_zeiss.log**

Файлы протоколов которые образуются и дополняются при работе приложения.

### **gcn\_monitor\_bta.bat** и **gcn\_monitor\_zeiss.bat**

Примеры bat-файлов для запуска приложения под *Windows*.

### **gcn\_proxy.py**

Специальная программа-ретранслятор VOEvent протоколов. Ждет обращения клиента. Затем сама вызывает один из трех GCN-серверов и организует параллельный процесс (thread) взаимной трансляции сообщений. Максимально возможна одновременная работа трех

клиентов, т.к. каждый GCN-сервер с этого IP-адреса примет только одно TCP-соединение.

## Диалоговые окна.

Поскольку запуск окна диалога блокирует вызывающую программу, управление ими оформлено как параллельный процесс (thread) чтобы не мешать процессу прослушивания сети. При старте этот thread показывает первое окно с информацией к какому GCN-серверу подключились.

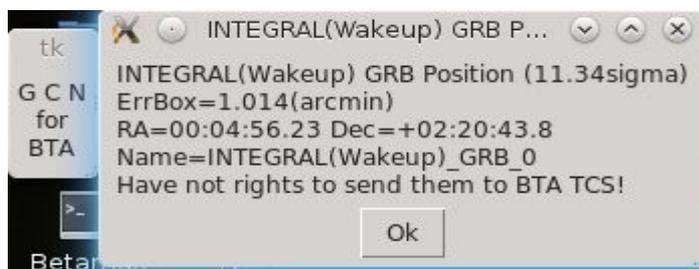


Бесконечно ждать реакции пользователя нельзя, поэтому для каждого окна запускается функция таймаута, которая уничтожает его по истечению времени.

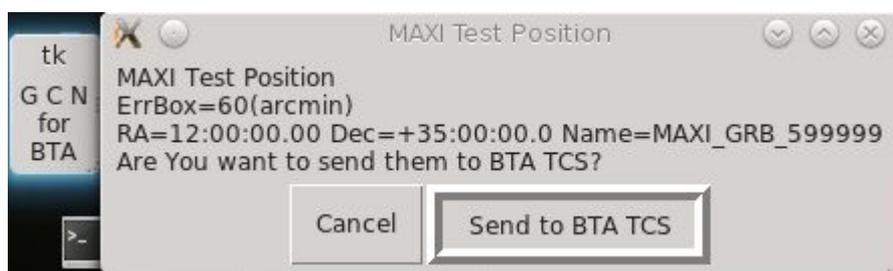
В каждый момент времени программа может показывать только одно окно требующее реакции наблюдателя. А сообщения после GRB-события могут идти сериями через секунды или минуты. Кроме того могут совпадать во времени сообщения от разных источников. Поэтому реализован следующий принцип приоритетности сообщений наблюдателю:

- новое предупреждение (warning) уничтожает предыдущее;
- сообщение о GRB уничтожает окно предупреждения;
- если выведено сообщение о GRB, предупреждения игнорируются;
- новое сообщение о GRB уничтожает предыдущее если его точность координат выше (ErrBox меньше), иначе отменяется.

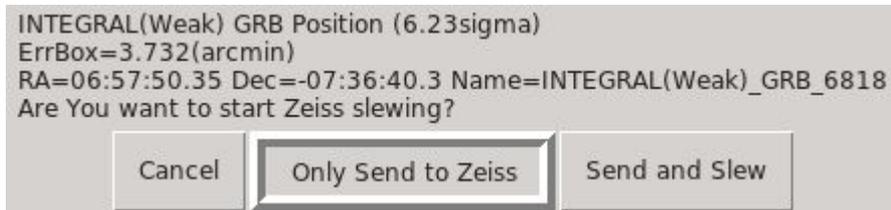
Окна сообщений о GRB выглядят по разному в зависимости от полученного уровня доступа к системе управления телескопа.



Доступа нет.

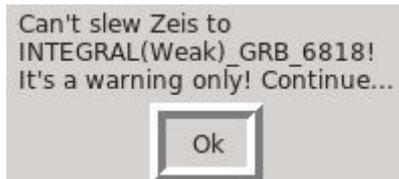


Только посылка координат в систему управления.



Можно сразу стартовать наведение телескопа.

Можно только послать координаты, затем в интерфейсе наблюдателя увидеть «куда попали», а наведение телескопа по введенным координатам запустить позже обычным образом. Если же сразу запустить наведение, то телескоп либо сразу поедет на объект, либо появится сообщение типа:

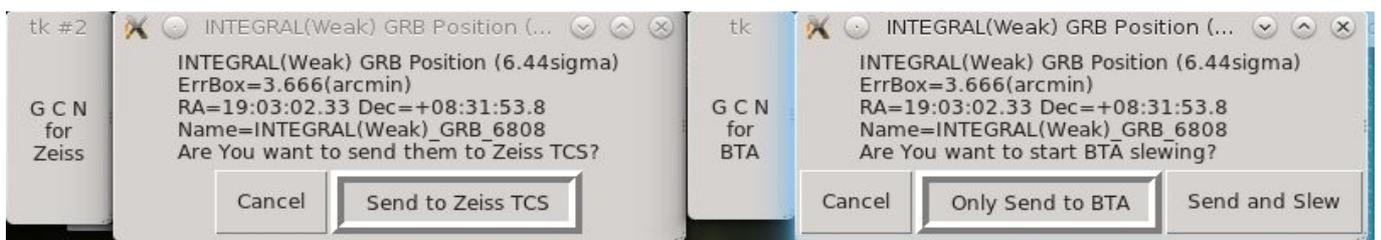


## Примеры работы.

Ниже приводятся примеры «дневных» испытаний но с реакцией на реальные GRB.

Программа-ретранслятор *gcn\_proxy.py* была стартована на сервере *www.sao.ru*. На клиентском компьютере стартованы сразу две программы *gcn\_monitor.py* с параметром *proxy=www.sao.ru*. Одна со 2-м уровнем (только координаты) для Цейсс-1000, другая с 4-м уровнем (наведение телескопа) для БТА.

Пришло сообщение от спутника INTEGRAL о GRB №6808. Видим на экране:



Сначала нажимаем «Send and Slew» для БТА.

В протоколе *gcn\_client\_bta.log* записано:

```
Thu Nov 13 11:12:00 2014 INTEGRAL(Weak) GRB Position (6.44sigma) message  
trigger: ID=6808 RA=19:03:02.33 Dec=+08:31:53.8, ErrBox=3.666' Signif=6.44sigma at 2014-11-  
13 08:11:55.8  
Thu Nov 13 11:17:50 2014 Fermi-GBM(Alert) just found a Transient or GRB 437559466  
(5.2sigma) at 2014-11-13 08:17:43.5  
Thu Nov 13 11:18:08 2014 Fermi-GBM(Flight) just found a Transient or GRB 437559466  
(RA=12:13:28.01,Dec=+79:01:59.9,ErrBox=15.1833degr,6.6sigma) at 2014-11-13 08:17:43.5  
Thu Nov 13 11:18:37 2014 Fermi-GBM(Ground) just found a Transient or GRB 437559466  
(RA=11:36:02.40,Dec=+79:40:48.0,ErrBox=9.64degr,0.8sigma) at 2014-11-13 08:17:43.5  
Thu Nov 13 11:30:00 2014 Slew telescope to RA=19:03:02.33 Dec=+08:31:53.8  
Name=INTEGRAL(Weak)_GRB_6808  
Coordinates were sent to BTA TCS
```

Start Slewing to new object - Ok

Т.е. сразу после этого пришла серия сообщений от Fermi GBM (Alert, Flight, Ground) о GRB №437559466, но они были проигнорированы из-за меньшего приоритета.

В протоколе АСУ БТА записано:

11:30:00.677 Компьютер: 192.168.2.85  
11:30:00.877 Вв: Alp=19 03 44.597 Del=+08 33 31.63  
11:30:00.877 A=-78 17 58.00 Z=+65 49 32.60  
11:30:01.177 Объект:INTEGRAL(Weak)\_GRB\_6808  
11:30:01.477 Старт наведения на объект  
11:30:01.978 Кнопка "Пуск" телескопа  
11:37:09.307 Переход в ведение по A/Z

Таким образом наведение на восток на Z~65° прошло успешно.

Также нажимаем «Send to Zeiss TCS» для Цейсс-1000. В интерфейсе наблюдателя на вкладке «Encdrs» видим что введенные координаты находятся в доступной области. Затем на вкладке «Object» нажимаем «GoToObject».

В протоколе *gcn\_client\_zeiss.log* записано:

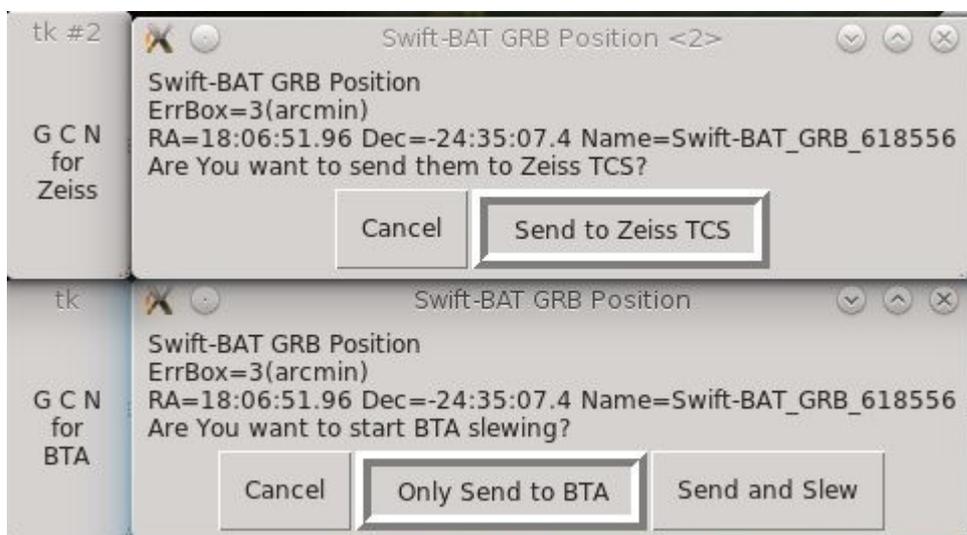
Thu Nov 13 11:30:06 2014 Send RA=19:03:02.33 Dec=+08:31:53.8  
Name=INTEGRAL(Weak)\_GRB\_6808 to Zeiss TCS  
RA=19.0506466667 Decl=8.5316 Epoch=2000.0 Name=INTEGRAL(Weak)\_GRB\_6808 - Ok

В протоколе системы управления Цейсс-1000 записано:

09:30:01.627 New User: "user" - Zeiss-1000 user (Set coord.only)(Level2)  
11:44:47.176 New User: "local" - Local(Level4)  
11:45:53.730 New Object: MeanRA=19:03:02.328 MeanDecl=+08:31:53.76 Epoch=2000.0  
Name="INTEGRAL(Weak) GRB 6808"  
LST=15:01:14.156 obsRA=19:03:40.228 ObsDec=+08:34:36.45  
HA=19:57:33.945 Az=+284:46:00.83 Z=+62:59:44.09  
11:45:54.130 Slew to object  
11:46:53.490 Slewing Ok! (dHA=0.01s dDecl=0.6") Switching to Tracking mode.

Также произошло успешное наведение на объект.

На следующей день появилось сообщение от Swift BAT о GRB №618556:



Скорее всего наведение невозможно ( $\delta \sim -24.5$ ), но пробуем отправить координаты в систему БТА.

В протоколе *gcn\_client\_bta.log* записано:

Fri Nov 14 11:02:07 2014 Swift-BAT GRB Position message

trigger: ID=618556 RA=18:06:51.96 Dec=-24:35:07.4, ErrBox=3' at 2014-11-14 07:32:04.39  
Fri Nov 14 11:02:22 2014 Swift-BAT Transient 618556, but not a GRB at 2014-11-14 07:32:03.21  
Fri Nov 14 11:09:36 2014 Send RA=18:06:51.96 Dec=-24:35:07.4 Name=Swift-  
BAT\_GRB\_618556 to BTA TCS  
Coordinates were sent to BTA TCS  
Warning! Object Z > 79degr!

Кстати, видим что через 15сек. пришло опровержение о том что событие №618556 уже не рассматривается как GRB.

На интерфейсе оператора видим что объект слишком низко, хотя часа через три его можно было бы наблюдать. В протоколе АСУ БТА записано:

11:09:36.801 Компьютер: 192.168.2.85  
11:09:37.001 Вв: Alp=18 07 45.880 Del=-24 34 49.38  
11:09:37.001 A=-48 13 33.99 Z=+84 40 59.17  
11:09:37.302 Объект:Swift-BAT\_GRB\_618556

На Цейсс-1000 координаты посылать не будем.