

Summer 7-3-2014

Accessing Information Using LVAAlert and Python

Belinda Cheeseboro

Andrews University, cheesebo@andrews.edu

Tiffany Summerscales Ph.D.

Andrews University, tzs@andrews.edu

Follow this and additional works at: <http://digitalcommons.andrews.edu/student-works>



Part of the [Physics Commons](#)

Recommended Citation

Cheeseboro, Belinda and Summerscales, Tiffany Ph.D., "Accessing Information Using LVAAlert and Python" (2014). *Posters, Presentations, and Papers*. Paper 24.

This Poster is brought to you for free and open access by the Undergraduate Research at Digital Commons @ Andrews University. It has been accepted for inclusion in Posters, Presentations, and Papers by an authorized administrator of Digital Commons @ Andrews University. For more information, please contact repository@andrews.edu.

Accessing Information Using LVAAlert and Python

B. D. Cheeseboro¹ & T. Z. Summerscales¹

¹Department of Physics, Andrews University
cheesebo@andrews.edu & tzs@andrews.edu

ABSTRACT

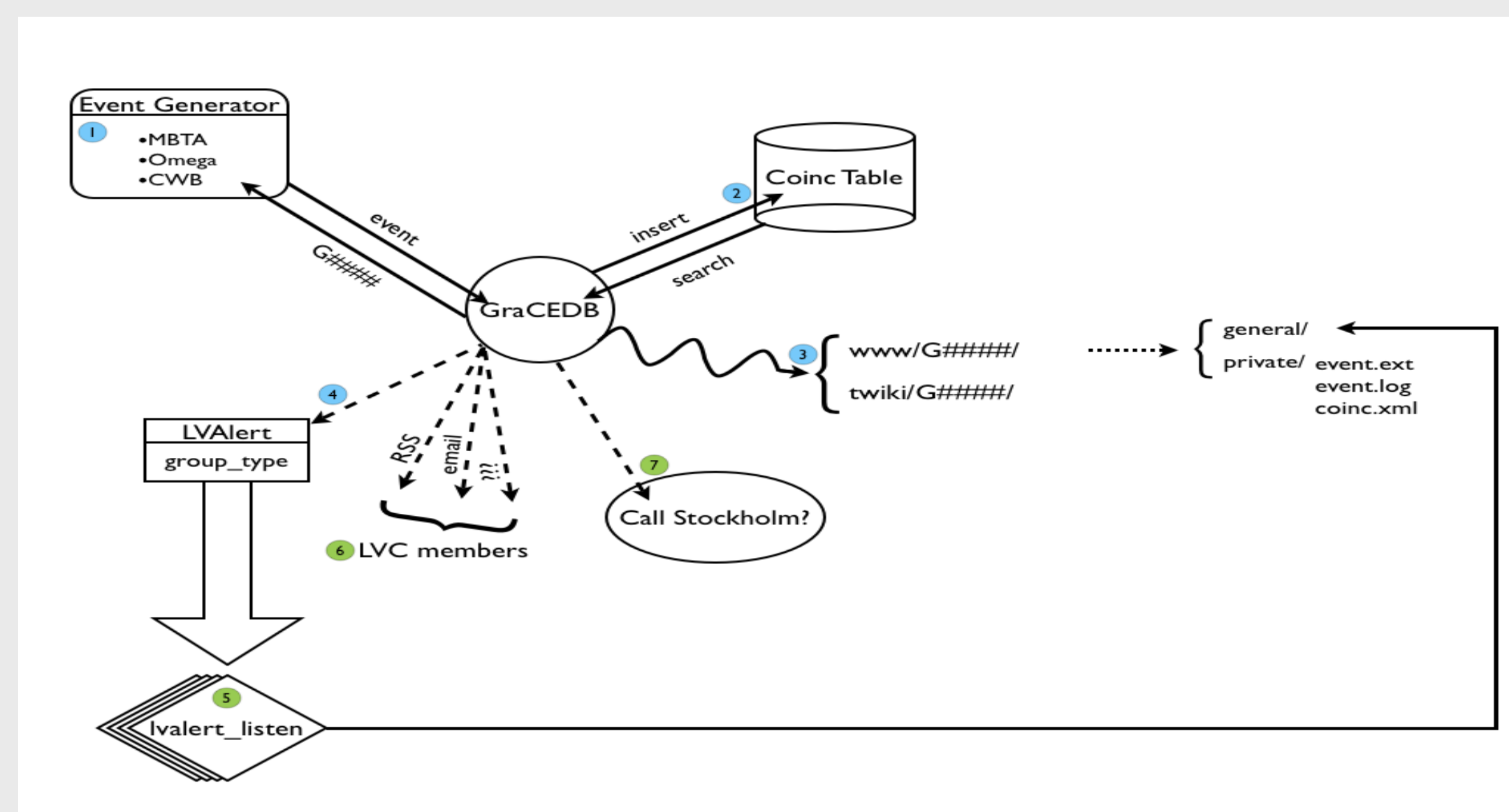
GraCEDb is a database that stores events that could possibly produce gravitational waves. By using the notification system, LVAAlert, we can develop a program in python to extract necessary information from that event and store it in a usable data structure. That data can then be analyzed by our algorithms to extract gravitational waveform data from the detectors

BACKGROUND

GraCEDb

Gravitational waves can be produced from two supermassive black holes coalescing into one, massive stars coming to the end of their life in a giant explosion called a supernova, and many other astronomical phenomena.

To keep track of all the possible events that are happening in our observable universe, we use the Gravitational wave Candidate Event Database (GraCEDb).



LVAAlert

To be notified an event has been entered into the database, we use the Ligo-Virgo Alert system (LVAAlert). This system automatically publishes the information to the node (channel) and those subscribed to that node are notified of the event.

METHODOLOGY

Python and XML

Python is a computer interpreted language that is excellent for parsing extensible markup language (xml) files. An xml file is a file that contains tags, elements, and attributes.

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<VOEvent xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:voe="http://www.lvoa.net/xml/VOEvent/v2.0"
xsi:schemaLocation="http://www.lvoa.net/xml/VOEvent/v2.0 http://www.lvoa.net/xml/VOEvent/VOEvent-v2.0.xsd" version="2.0" role="observation"
ivorn="lvo://nasa.gsfc.gov/Fermi4GBM_Alert_2014-02-26T19:03:14.34_415134190_1-229">
  <Who>
    <AuthorIVORN>lvo://nasa.gsfc.gov/gcn/AuthorIVORN</AuthorIVORN>
    <Date>2014-02-26T19:03:14</Date>
  </Who>
  <Description>
    This VOEvent message was created with GCN VOE version: 1.12 18Feb14
  </Description>
  <Author>
    <shortName>Fermi (via VO-GCN)</shortName>
    <contactName>Julie McEnery</contactName>
    <contactEmail>Julie.E.McEnery@nasa.gov</contactEmail>
    <contactPhone>+1-301-286-1632</contactPhone>
  </Author>
  <Rat>
    <Param name="Packet_Type" dataType="string" value="110"/>
    <Param name="Pkt_Ser_Num" dataType="string" value="1"/>
    <Param name="TrigID" dataType="string" value="415134190" ucd="meta.id"/>
    <Param name="Sequence_Num" dataType="string" value="1" ucd="meta.id.part"/>
    <Param name="Burst_T0D" dataType="string" value="16714" ucd="time" unit="days"/>
    <Param name="Burst_S0D" dataType="string" value="66587.34" ucd="time" unit="sec"/>
    <Param name="Trig_Signif" dataType="string" value="9.4" ucd="stat.sig" unit="sigma"/>
    <Param name="Trig_Dur" dataType="string" value="0.016" ucd="time.interval" unit="sec"/>
    <Param name="Lo_Chan_Index" dataType="string" value="4" unit="dn"/>
    <Param name="Hi_Chan_Index" dataType="string" value="7" unit="dn"/>
    <Param name="Lo_Chan_Energy" dataType="string" value="98" unit="keV"/>
    <Param name="Hi_Chan_Energy" dataType="string" value="infinity" unit="keV"/>
    <Param name="ADC_Lo_Chan" dataType="string" value="497" unit="dn"/>
    <Param name="ADC_Hi_Chan" dataType="string" value="1995" unit="dn"/>
    <Param name="Algorithm" dataType="string" value="50"/>
    <Param name="Data" dataType="string" value="0x0"/>
    <Param name="Misc_flags" dataType="string" value="0x0"/>
  </Rat>
</VOEvent>
```

The data of an event candidate is stored in xml files. So by writing a python program to access the information that we need from those files. Then we can use that information in algorithms to analyze the data from that event

PROGRESS MADE

So far I have signed up to receive alerts from LVAAlert, retrieved data from the GraCEDb website, and currently working on parsing xml data files from GraCEDb.

WHAT'S NEXT?

The next step after parsing xml files is working with LVAAlert to accessing the information from GraCEDb directly using terminal.

REFERENCES

- LIGO GraCEDb Wiki page
<https://www.lsc-group.phys.uwm.edu/daswg/wiki/HowtoGraceDb>
- LIGO Data Analysis Software Group
<https://www.lsc-group.phys.uwm.edu/daswg/projects/lvalert.html>

ACKNOWLEDGEMENTS

LIGO research at Andrews University is supported by NSF grant PHY-0969810

