# Gallifray: A VLBI Geometric Modelling and Parameter Estimation Framework for Black hole images using Bayesian Techniques


**Saurabh* , Sourabh Nampalliwar****

*Dyal Singh College, University of Delhi, Lodhi Rd, New Delhi, India
**Theoretical Astrophysics, Eberhard Karls Universität Tübingen, Tübingen, Germany
E-mail: sbhkmr1999@gmail.com

**PRESENTED AT:**

## WHAT IS IT?

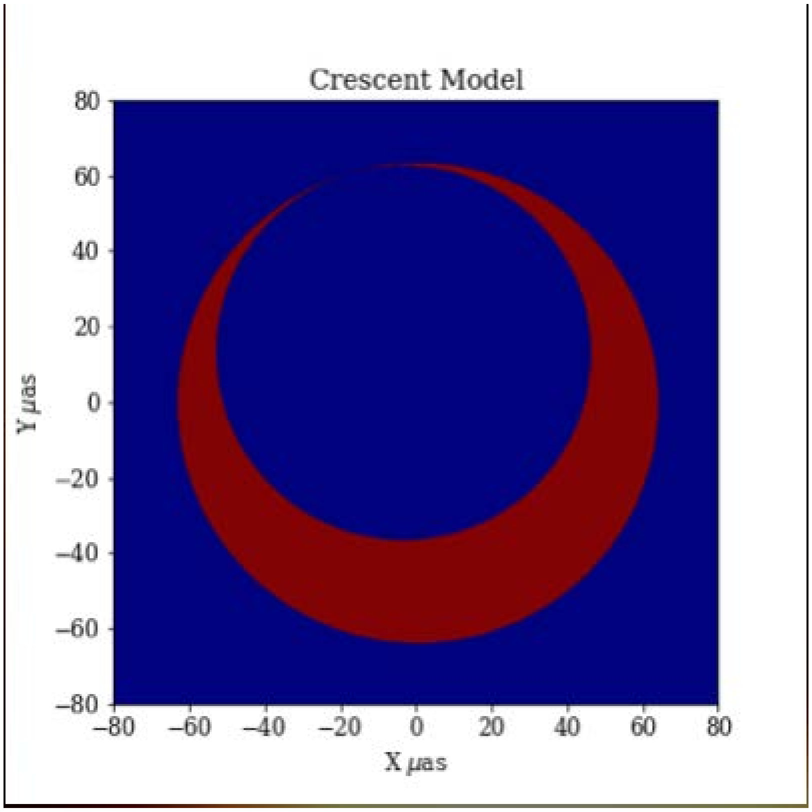
- Modelling tool [Geometric + General Relativistic]
- Extracting basic parameters (Diameter, width, etc.)
- Bayesian formalism and parameter extraction
- Written purely in Python
- Code available at github

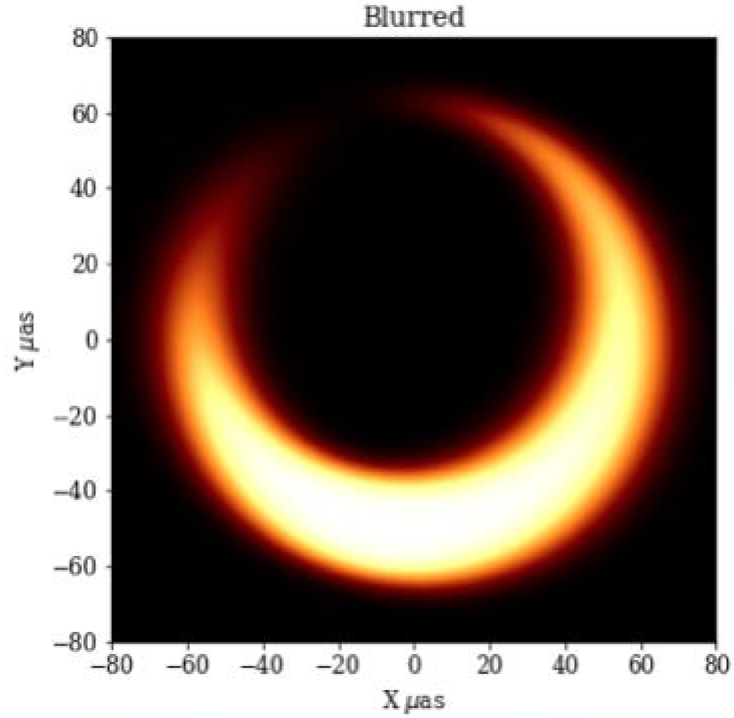
https://github.com/Relativist1/Gallifray
(https://github.com/Relativist1/Gallifray)

# VISIBILITY MODELLING (GEOMETRIC MODELS)

Simplification of a black hole accretion image.

Eg: Crescent Model*




Blurred
Y μas
X μas
80
60
40
20
0
−20
−40
−60
−80
−80
−60
−40
−20
0
20
40
60
80

## MODEL FITTING
## BASIC RECIPE (VIS. AMP.)

- Choose a Quantity to fit: Visibility Amplitude (for starters)
- Using the correlated data sets + time-averaged
- Construct the likelihood function. eg: Gaussian

$$\mathcal{L}(V_{ij}) = e^{-\chi^2/2}$$

where

$$\chi^2 = \Sigma_{i=1}^{n_d} \frac{(\mathcal{V}_{R_i} - \mathcal{V}_{mR_i})^2 + (\mathcal{V}_{I_i} - \mathcal{V}_{mI_i})^2}{\sigma_i^2}$$

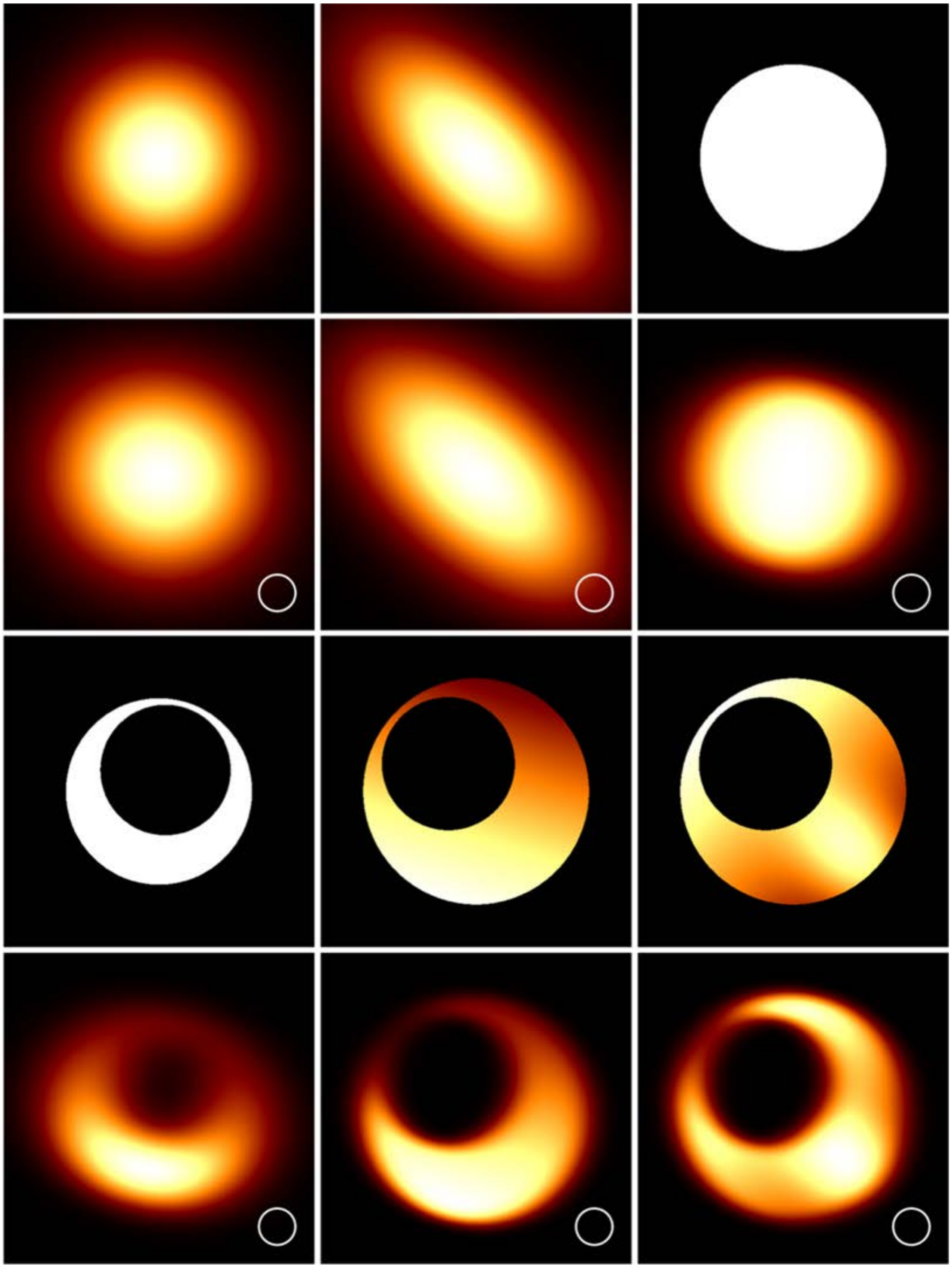
$$\mathcal{V} \rightarrow \text{Observed Visibilities}$$

$$\mathcal{V}_m \rightarrow \text{Model Visibilities}$$

# IMPLEMENTATIONS

## Models

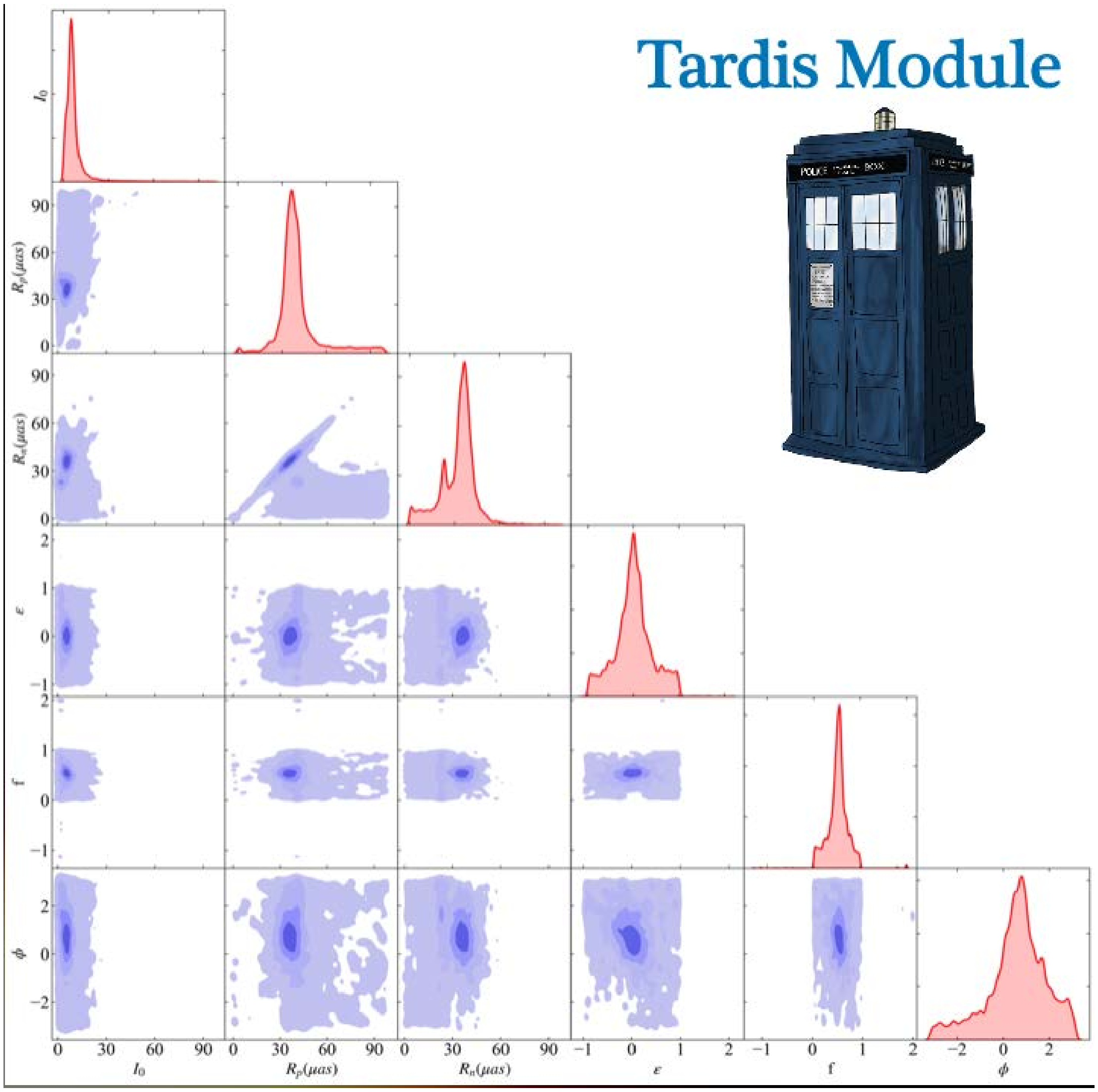
- Symmetric Gaussian
- Asymmetric Gaussian
- Filled Disk
- Crescent
- xs-ring
- xs-ringauss



Plotting Scheme

- TARDIS (Triangular Posterior Distribution Plotting)

# Tardis Module



# MODEL FITTING (MCMC)
# LIKELIHOOD CONSTRUCTION

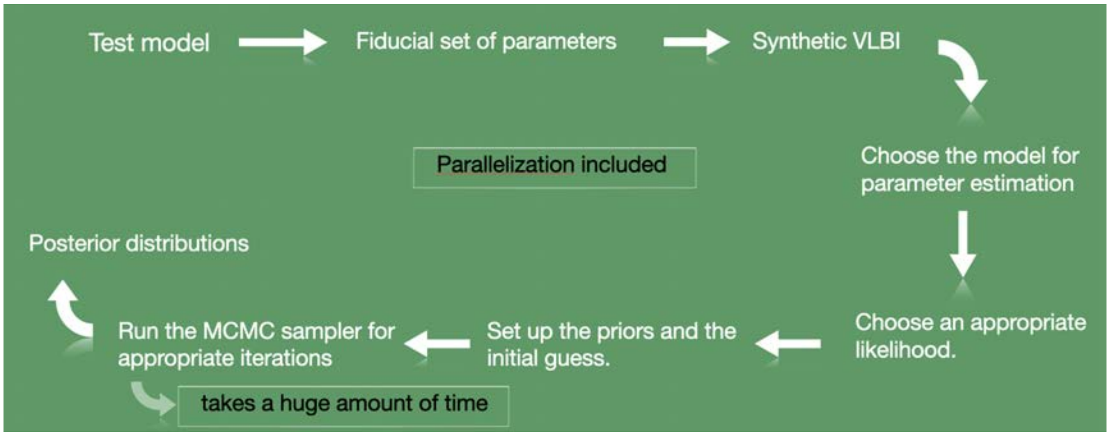
- Defining the Posterior Likelihood distribution, for Visibility Amplitude, the aptest is the Rice Distribution

$$P^{obs}(\mathcal{V}_m; \mathcal{V}_{ij}, \sigma_{ij}) =$$
$$\frac{\mathcal{V}_m}{-\sigma_{ij}^2}\left[\frac{-(\mathcal{V}_m^2+\mathcal{V}_{ij}^2)}{2\sigma_{ij}^2}\right] I_0\left(\frac{\mathcal{V}_m\mathcal{V}_{ij}}{\sigma_{ij}^2}\right)$$
$$I_0 \rightarrow \text{Modified Bessel function of the first kind}$$

- at high S/N (S/N 2), this becomes a gaussian.

# WORKFLOW AND FUTURE ADDITIONS

## Workflow




## Eg: Model image reconstruction (using eht-imaging)

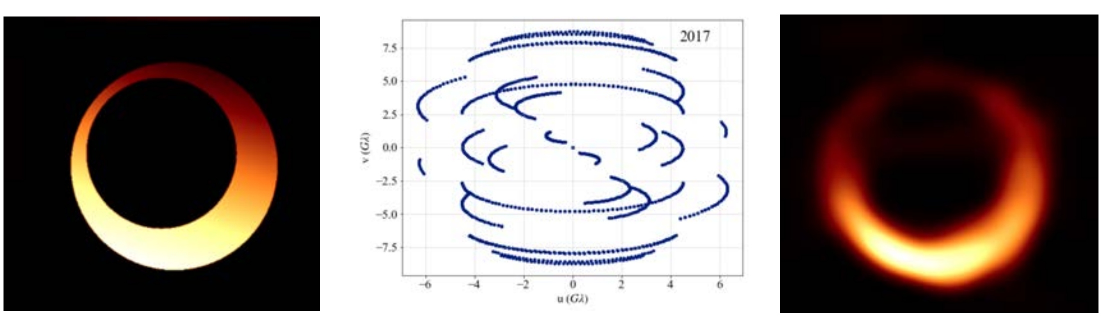


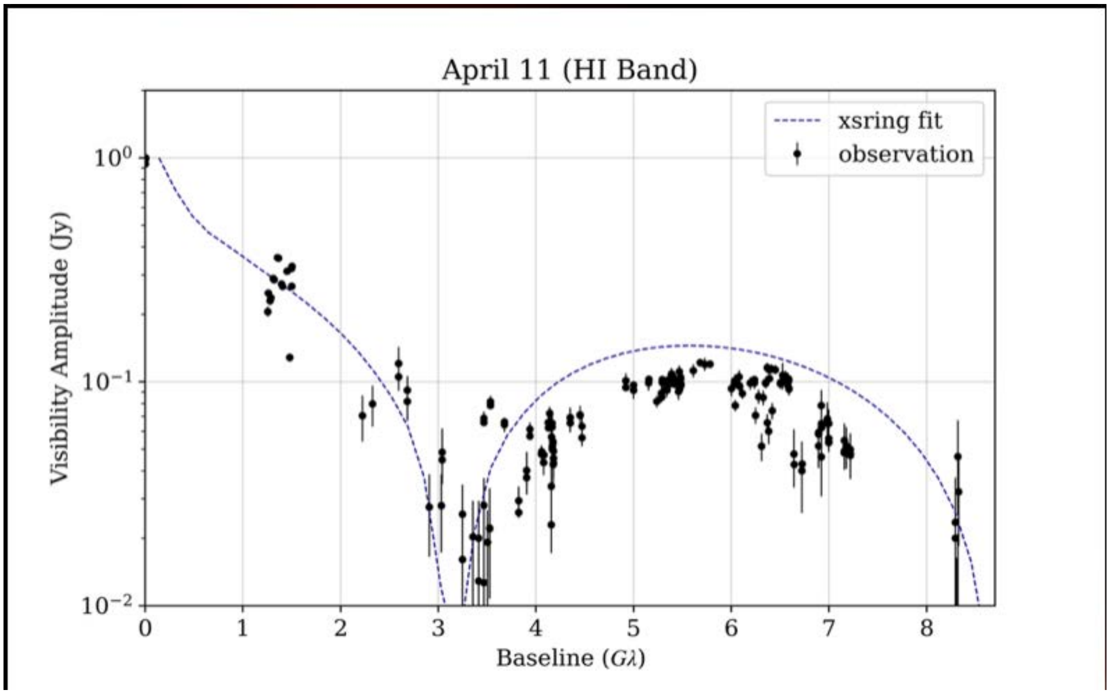
Fig: left to right - Model Image, Baseline Coverage (EHT-2017 config), Reconstructed Image

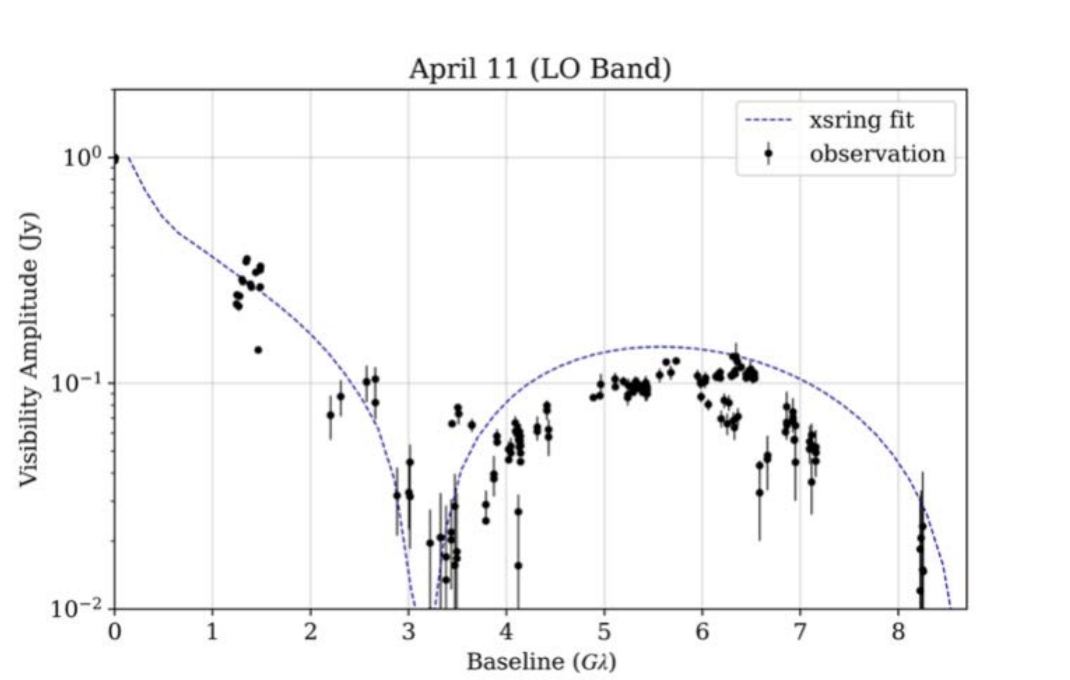
## Future Additions

- Ray-Tracing Capability (2nd Release)
- Static and Spherically Black hole surrounded with spherical accretion
- Nesting Sampling for MCMC
- Additions of more models (modeling jets etc.)
- Embedded ray-tracing routines with different accretion models.
- Space-VLBI

# M87 FITS (XSRING MODEL)

- Model fitting for 2017 M87 results (Visibility Amplitudes)




-

# AUTHOR INFORMATION

Saurabh* , Sourabh Nampalliwar**

*Dyal Singh College, University of Delhi, Lodhi Rd, New Delhi, India

**Theoretical Astrophysics, Eberhard Karls Universität Tübingen, Tübingen, Germany

## ABSTRACT

Recent observations from the EHT of the centre of the M87 galaxy has opened up a whole new era for testing general relativity using BH (Black hole) images generated from VLBI. While different theories have their version of BH solutions, there are some 'geometric models' as well which can be approximated to visualise the image of a BH in addition to understand the geometric properties of the radio source such that ring size, width, etc. To incorporate and implement such framework, different methods and techniques are needed to be explored for doing such model comparison. We present 'Gallifray' [1], an open-source Python based framework for geometric modelling and estimation/extraction of parameters. We employ Bayesian techniques for the analysis and extraction of parameters. In my presentation, I will talk about the workflow, preliminary results obtained and applications of the library for image/model comparison. I will also talk about the scope of the library in testing Black hole images for any possible deviation from Kerr spacetime.

References:

[1] https://github.com/Relativist1/Gallifray/

## REFERENCES

https://github.com/Relativist1/Gallifray/