# SysML model of Exoplanet Archive Functionality and Activities

Solange Ramirez*[a], and the NASA Exoplanet Archive Team [a]
[a] NASA Exoplanet Science Institute, California Institute of Technology, 1200 East California Blvd.
Pasadena, CA USA 91125

## ABSTRACT

The NASA Exoplanet Archive is an online service that serves data and information on exoplanets and their host stars to help astronomical research related to search for and characterization of extra-solar planetary systems. In order to provide the most up to date data sets to the users, the exoplanet archive performs weekly updates that include additions into the database and updates to the services as needed. These weekly updates are complex due to interfaces within the archive. I will be presenting a SysML model that helps us perform these update activities in a weekly basis.

**Keywords:** Model-based System Engineering, SysML, Exoplanet, Archive.

## 1. INTRODUCTION

The NASA Exoplanet Archive[1] is a database and toolset funded by NASA and available on line to support astronomical research related to exoplanet science. The archive combines a database of explanet and planet host parameters and key public data sets from space and ground-based observatories. Data included are planet parameters such as mass, radii, and equilibrium temperatures; orbital information such as periods, semi-major axis, eccentricity; transit parameters such as transit duration, mid-point, ingress time, stellar parameters for the host such as positions, astrometry, photometry, stellar properties (effective temperature, surface gravity, stellar mass and radii, metallicity), and planet discovery information. All values available in the archive are linked to the corresponding publications providing traceability to our users. The archive also incorporates more than 20 million light curves from space missions such as *Kepler* [2] and CoRoT[3] and from ground-based surveys such as SuperWASP[4], KELT[5], and others. Our goal is to provide a single location and a consistent toolset to facilitate exoplanet research with the most up to date information available.

The NASA Exoplanet Archive has a process in place to add data about newly discovered planets into the archive. The exoplanet data (parameters, light curves, images, etc.) come from refereed publications members of our team monitor continuously to identify new discoveries and updated information arising from new astronomical observations. This process has been incorporated into a SysML (Systems Modeling Language) [6] model that provides details on the structure and behavior of the archive in order to facilitate the related updates of individual services within the archive.

Aspects of the Exoplanet Archive SysML model will be presented in this article: section 2 gives an overview of SysML and the tool used for the model; section 3 describes a high level structure of the archive and the internal database; section 4 explains the actors involved and the weekly update process; section 5 shows an example on the incorporation of a set of planets discovered by the radial velocity (RV) technique; and section 6 discusses future expansion of this model.

## 2. SYSML OVERVIEW

The Systems Modeling Language (SysML) is a general-purpose visual modeling language for application to model-based systems engineering. It is derived from the Unified Modeling Language (UML)[7], it reuses parts of UML and adds extensions specific for systems engineering usage. SysML does not impose a particular method of engineering, in fact SysML supports traditional structured analysis (functional decomposition and allocation of functions to components) as well as use case driven approach (scenario analysis). SysML provides nine different types of diagrams, grouped into three sets:

1. The requirements diagram is the only type in the requirement set. Requirements can be represented in a text-based form as well as in a graphical form. The graphical form has the ability to show relationships among requirements and other elements of the model.

[*] solange@ipac.caltech.edu

Modeling, Systems Engineering, and Project Management for Astronomy VII,
edited by George Z. Angeli, Philippe Dierickx, Proc. of SPIE Vol. 9911, 991129
© 2016 SPIE · CCC code: 0277-786X/16/$18 · doi: 10.1117/12.2232862

2. Structural Diagrams represent aspects of the logical and physical architecture of the system. The diagrams in this set are: Package, Block Definition, Internal Block, and Parametric.

3. Behavioral Diagrams describe usage and functionality of the system. The diagrams in this set are: Use Case, Activity, Sequence, and State Machines.

In the model presented here, I will use Block Definition, Internal Block, Use Case, and Activity diagrams to represent aspects related to the weekly update performed by members of the Exoplanet Archive team in order to make available the most up to date data of exoplanets discovered and characterized so far. The model presented here has been made using the Paryrus[8] tool, which is part of the Eclipse environment.

## 3. EXOPLANET ARCHIVE STRUCTURE

The high level structure of the Exoplanet Archive is shown in Figure 1, which is a Block Definition Diagram. The Exoplanet Archive has three basic operations (gear symbol) defined within its block. These operations are related to different types of searches performed by a user: search for a single object, find inventory information, and access data for a group of objects. These operations can be performed manually by a user or automatically by a script.

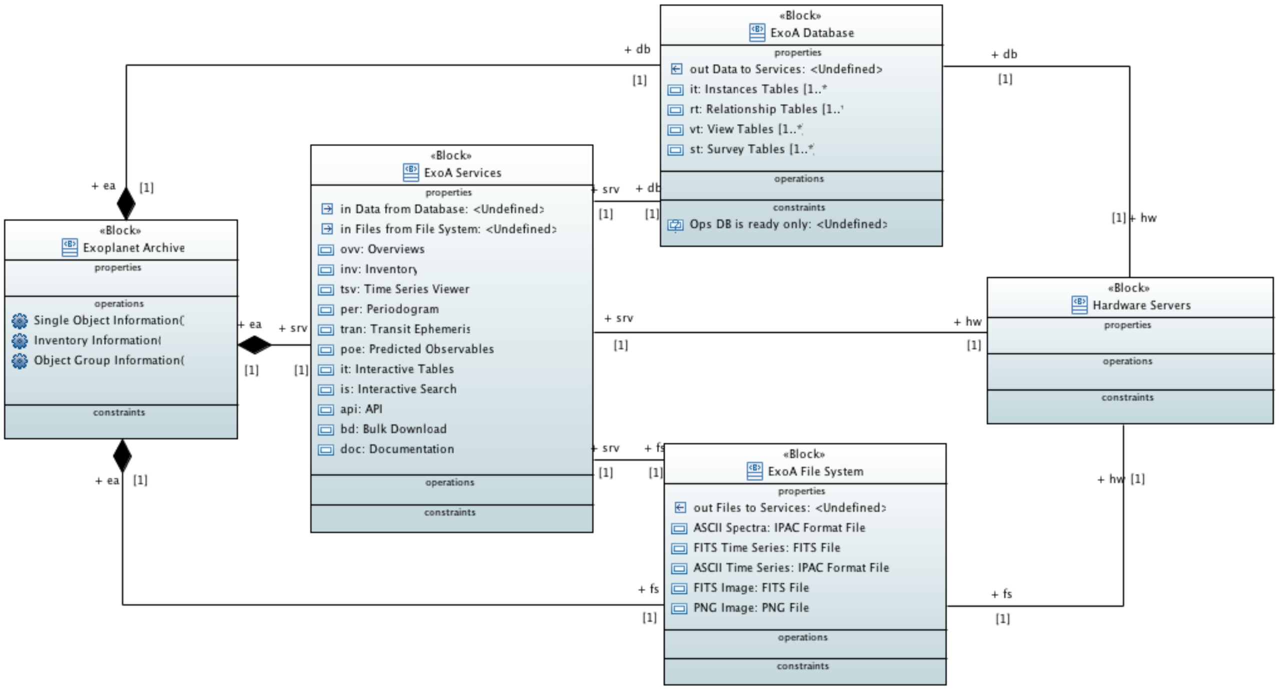

**Figure 1: Block Definition Diagram showing the high level structure of the Exoplanet Archive**

There are three basic components of the archive: Database, Services, and File System, all of them have respective hardware servers. Four types of tables, which are Instances, Relationship, View, and Survey, compose the database (see also Figure 2).

The File System contains Time Series, Spectra and Images in different formats (ASCII, FITS, and PNG files) and they are shown as parts (rectangular symbol) within the *ExoA File System* block in Figure 1.

The Services provide different functionality to the users, by combining data retrieved from the database and files accessed from the file system. The following services are also shown as parts (rectangular symbol) in the *ExoA Services* block in Figure 1:

- Overviews: lists different sets of parameters related to a single object. All the parameter values are associated to the corresponding refereed publications.
- Inventory: provides a summary of available information found in the archive within a user provide radius around a location in the sky given by an object name or position.

- Time Series viewer: visualizes time series and allows the user to graphically interact with the data.
- Periodogram: extracts periodic signals from time series data.
- Transit Ephemeris: calculates when transits and orbital phase quadrate will happen and when they can be observed at a particular ground-based observatory or space facility.
- Predicted Observables: provides estimates of the Habitable Zone (HZ) parameters and predicted planet signatures (e.g. RV and astrometric wobbles, transit depths, etc.) for stellar objects in the archive or hypothetical stars.
- Interactive Tables: display data currently stored in the archive and allows the user to search, filter, sort, plot and download all of parts of the data within a particular table.
- Interactive Search: allows users to search for and identify subset(s) of large data sets to view in an interactive table.
- API: provides access to our data by scripts that can perform specific and well-documented search queries.
- Bulk Download: allows users to download data on bulk using pre-generated scripts.
- Documentation: provide general and detailed information about the archive, its holding, and user functionality.

Four types of tables, shown as parts in within the *ExoA Database* block, compose the database:

- Instances Tables: store parameters or sets of parameters associated to an object in the sky (stars, planets, planet candidates)
- Relationship Tables: define relationships between objects in the sky (stars, planets, planet candidates).
- View Tables: are built from types of objects are their corresponding parameters.
- Survey Tables: store metadata associated to time series from ground-based surveys and/or space missions.

Ports, defined in SysML as rectangles with arrows, describe ways the blocks are connected with each other. The ports defined in the blocks in Figure 1, show that the ExoA Services use data and files from the database and file system, respectively.

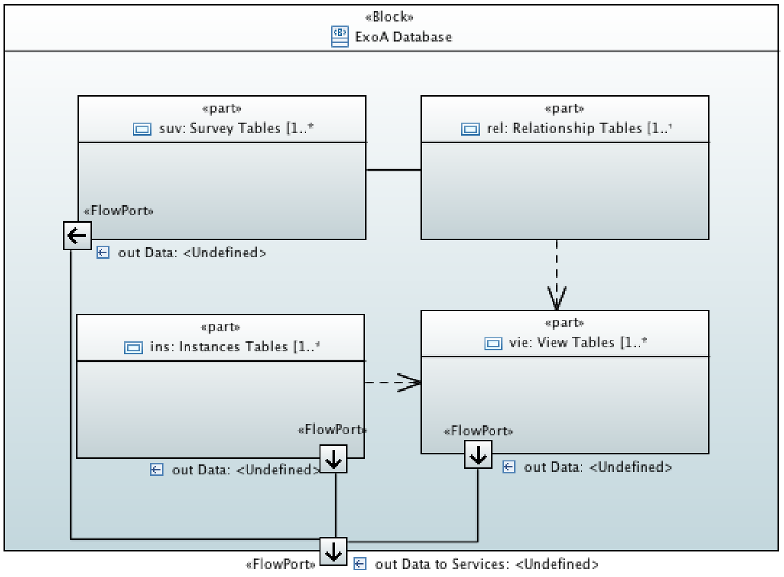

**Figure 2: Internal Block Diagram showing the relationship of the parts within the ExoA Database.**

Figure 2 shows the internal block diagram representing how the four types of tables in the *ExoA Database* block relate to each other. The dotted arrows represent dependencies between the parts. In the case of Figure 2, the dependency indicates that object information from the Relationship Tables and parameter information from the Instances Tables form the content of the View Tables.

Figure 2 also show more details of the flow of data to the *ExoA Services*, which access data from all types of tables with the exception of data from the Relationship Tables.

## 4. WEEKLY UPDATE PROCESS

The NASA Exoplanet team performs a weekly update of data whenever there are new exoplanets discovered in the published literature or new information (for example: parameters and time series data) on already confirmed planets. This activity is crucial to fulfill the goal of providing the most up to date exoplanet information to the astronomical community. Members of our team participate with different roles to perform the weekly update activities. The roles (in our team a person can have more than one role) involved in this process are: Archive Scientist, Data Analyst, Ingestion Operator, System Administrator, Manager, and Archive User. Figure 3 shows the different responsibilities for each role in the form of a Use Case Diagram: roles are represented by stick figures; responsibilities or actions are represented by oval elements.

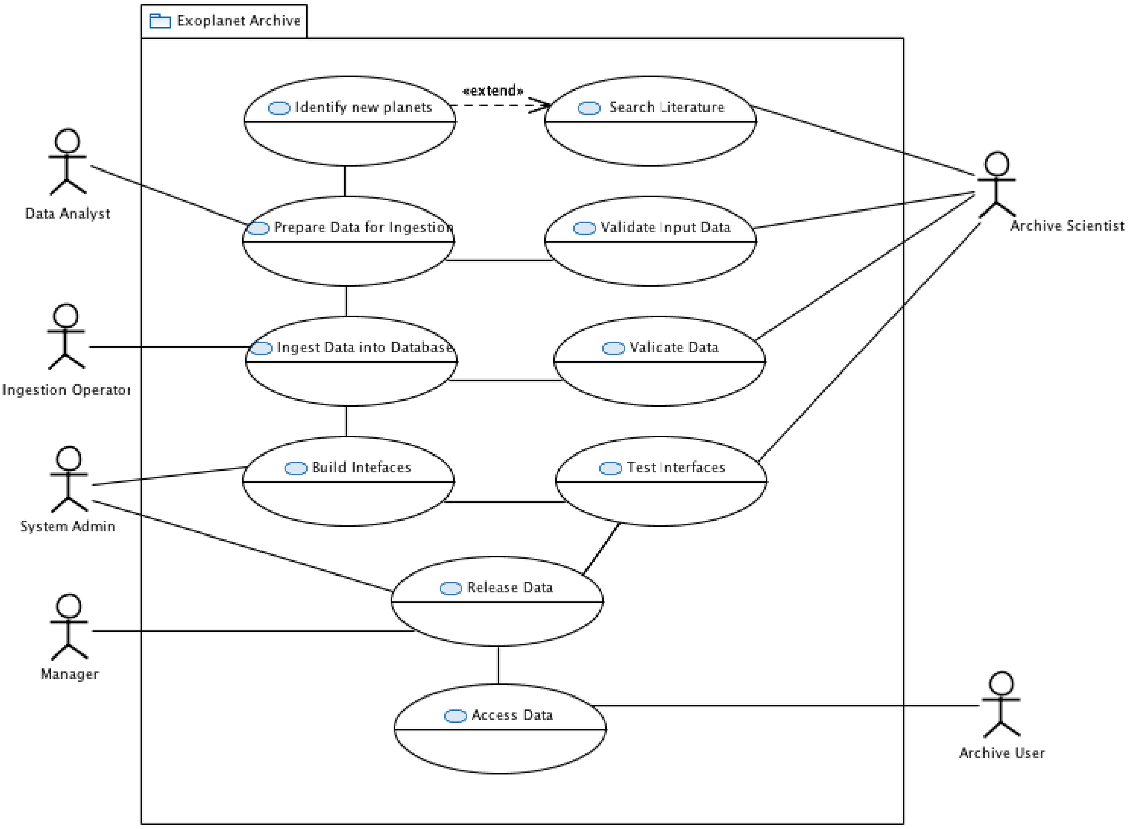

**Figure 3: Use Case Diagram representing action my team roles in the context of the weekly update activities**

The Use Case diagram portraits how the system is used. In the case of Figure 3, our team members perform most of the actions and roles internally, with the exception of *Access Data* that can be played by any archive user.
Some of the actions in the diagram are related: connected by a line (simple association) or by a dotted arrow (extend relationship). The extend relationship between *Search Literature* and *Identify new planets* convey the possibility that there maybe new planets identified after a literature search.

Another form to describe the weekly update process is through an Activity Diagram, which portraits order of actions, transformation of inputs and outputs, and control, as shown in Figure 4.

In an Activity Diagram, actions are represented by round cornered rectangles, decision points as diamonds, join and fork nodes as thin rectangles. The arrows denote the flow of the action, starting at an initial node (black dot) and finishing at a flow final node (crossed white circle) or at an activity final node (black dot on white circle). The Activity Diagram in Figure 4 also shows estimated times in the week for some of the actions to be completed in order to meet the nominal deadline for the new data to be released every week on Thursdays at 10am.

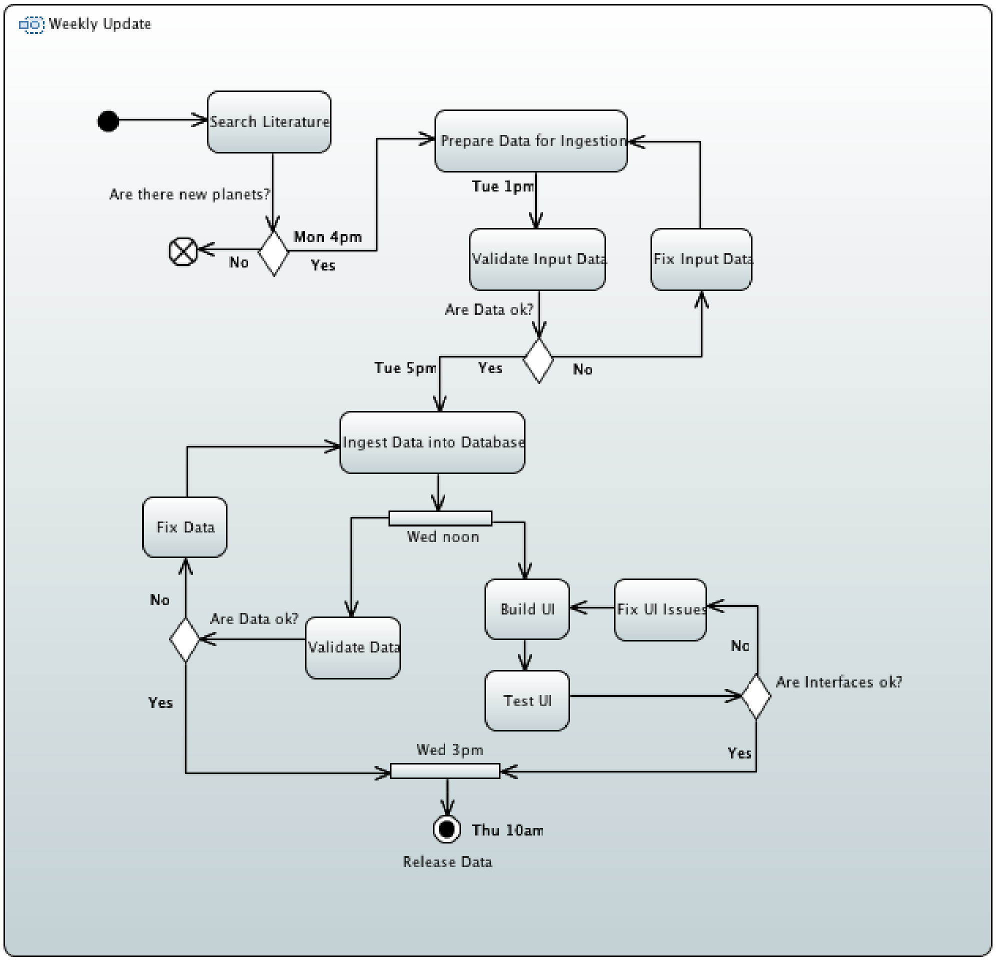

**Figure 4: Activity Diagram describing the weekly update process.**

## 5. NEW PLANETS EXAMPLE: FROM RV TECHNIQUE

One of the uses of this model is to visually represent the impact of new data into the services we provide to our users. The visual representation helps us as a team to streamline the validation of the new data and the testing of the user interface (UI) parts or services that are dependent on the new data. The simplest example of new data is new planets discovered by the radial velocity (RV) technique. These planets are associated to relatively bright stars that rarely overlap in sky position with the field of view of the space mission which data we serve in the archive. This last assumption will break when data from future all sky space missions become available. This model will be instrumental in the incorporation of new data sets into the archive.

Figure 5 shows the dependencies from the point of view of the database as SysML Internal Block Diagram. Elements internal to the database are shown as parts (solid rectangles) and elements external to the database are shown as references (dotted line rectangles). The element dependencies have a sense of a flow of data, starting with the *Ingestion Tool* that allows data to be added into the database, continuing with the population of views (*Confirmed Planets* and *Multi Exo Pars*), and finishing with the services that need to be built and/or tested to verify that the new data content is accessible by users. The ports in the *Ingestion Tool* and database tables represent the flow into the database of the data associated to the new RV planets. The relationship between database tables (*Object Relationship, Planet Relationship,* and *Instances Tables*) and the views (*Confirmed Planets* and *Multi Exo Pars*) is a dependency named *populate* that indicates that the views are constructed from the data content of the database tables. There are two more relationships. The first is a dependency named *build & test* that denotes that the added RV planet data need to the built into the linked services. The second is a connection names *test* that indicates that the added RV planet data can be verified as present in the linked service.

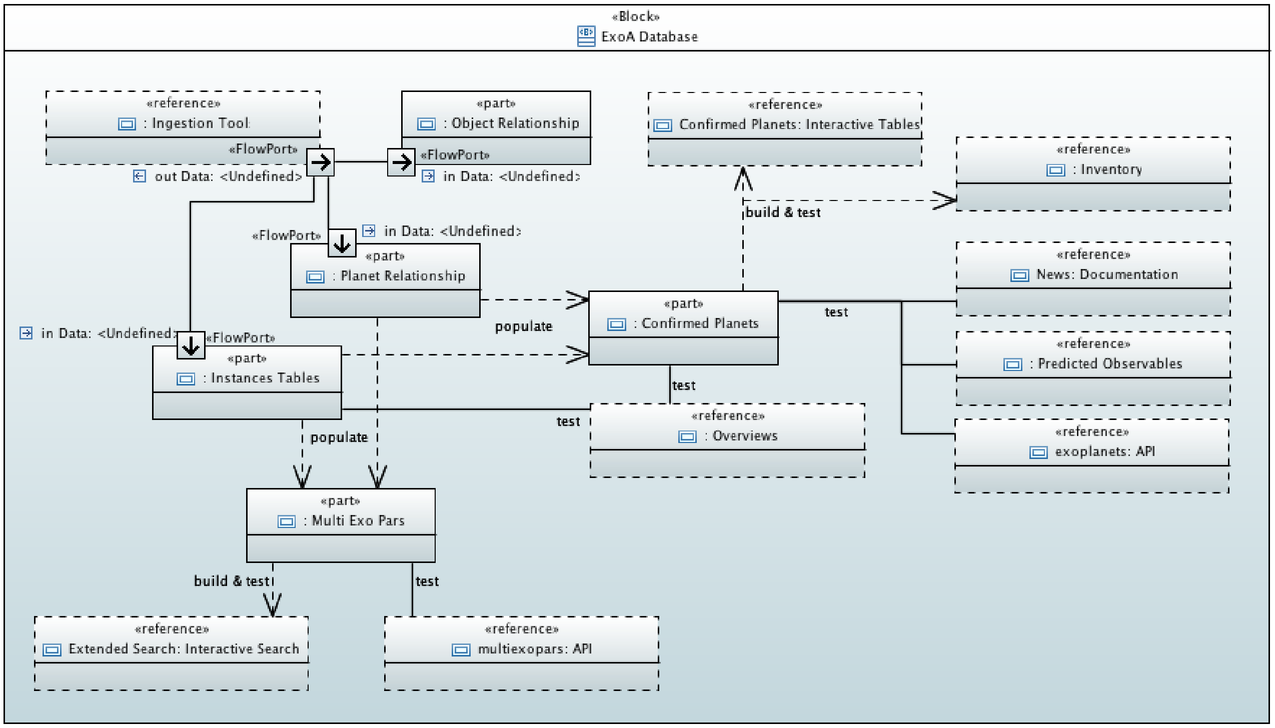

**Figure 5: Internal Block Diagram showing dependencies between elements of the database and archive services due to the incorporation of data on planets discovered using the RV technique.**

## 6. FUTURE EXPANSION OF THIS MODEL

The model presented here is an initial effort that will continue to grow in the future. Next, we will to expand the number of diagrams to include the dependencies between model elements due to the addition of planets discovered other techniques, that related to additional database tables (*Survey Tables* for example) and additional services (*Time Series Viewer, Periodogram, Bulk Download*). At present, diagrams developed in this model have proven crucial in the daily activities of the team and in the training of newly hired team members.

## REFERENCES

[1] Akeson, R., et al., "The NASA Exoplanet Archive: Data and Tools for Exoplanet Research", PASP, 125, 989 (2013).

[2] Borucki, W. J., Koch, D. G., Basri, G., et al., "Kepler Planet-Detection Mission: Introduction and First Results", Science, 327, 977 (2010)

[3] Auvergne, M., Bodin, P., Boisnard, L., et al., "The CoRoT satellite in flight: description and performance", A&A, 506, 411 (2009)

[4] Butters, O. W., West, R. G., et al., "The first WASP public data release", A&A, 520, 10 (2010).

[5] Pepper, J., Pogge, R. W., DePoy, D. L., et al., "The Kilodegree Extremely Little Telescope (KELT): A Small Robotic Telescope for Large-Area Synoptic Surveys", PASP, 119, 923 (2007).

[6] OMG, "OMG Systems Modeling Language", (2016) < http://www.omgsysml.org/>

[7] OMG, "Unified Modeling Language", (2016) < http://www.uml.org/>

[8] The Eclipse Foundation, "Papyrus Modeling Environment", (2015) < https://eclipse.org/papyrus/>