UML diagrams for the AliEn
job execution part and PackMan service

Armenuhi Abramyan

A.I. Alikhanyan National Scientific Laboratory

aabramya@mail.yerphi.am

Outline

- **AliEn Services**

- Demand for improvement of AliEn services

- **Unified Modeling Language Sequence diagrams**

- How the diagrams are created

    - Installing the UML::Sequence tool

    - Converting the Perl code to the text format

    - Passing created file to the UML::Sequence tool

- **How to read the generated diagram**

- Sequence diagrams for AliEn job execution part

- **Sequence diagrams for AliEn PackMan service**

- Conclusion

- **Acknowledgements**

AliEn Services

The core components of **AliEn** infrastructure are Web services, which communicate with each other via XML-based messages using **SOAP** (Simple Object Access Protocol) protocol. The major part of AliEn (about 95%) is constructed from the components of a large number of the common Open Source software (such as Globus/GSI, OpenSSL, OpenLDAP, SOAPLite, MySQL, Perl). The native AliEn code is written predominantly in **Perl**.

The AliEn services can be divided into 2 categories:

Central services:

Application Programming Interface,
Authentication,
Job Manager, Transfer Manager,
Job Broker, Transfer Broker,
Task Queue and File Catalogue databases,
Transfer Optimizer,
Catalogue Optimizer,
Job Optimizer,
FTD,
MonALISA,
Logger and Information Service.

Site services:

Computing Element,
Cluster Monitor,
Package Manager,
MonALISA,

Running on Worker Node:

Job Agent

Demand for improvement of AliEn services

Starting with the commissioning of **ALICE** detector in the end of 2009 and experimental data taking and analysis, one has been observing a very rapid growth of the number of submitted jobs requiring large computing time and resources run on **AliEn**. This growth has put AliEn under pressure of extreme loads, which revealed an inadequate reaction of some of its services, such as **PackMan** and **File Catalogue**.

These services need to be reworked and optimized to sustain the high load.

Transparent view of the AliEn functionality

In order to support the continuous process of the improvement of AliEn services and facilitate the effective involvement of new developers, **ALICE Offline group** has decided to provide them with a transparent view of the AliEn functionality.
This is done by means of diagrams describing the dynamic behaviour of AliEn services, their interrelationship.

Currently the diagrams for the following operations are available:

- File transfers - Lola Saiz Santos

- Job execution – Armenuhi Abramyan (A. A.)

- Experiment software packages installation and configuration (PackMan) – A. A.

- Basic Catalogue functions (directory creation, file registration and removal, etc) – A. A.

  http://alien2.cern.ch/index.php?option=com_content&view=article&id=84%3Adeveloper-main&catid=7&Itemid=64

The diagrams are represented in **Unified Modeling Language**

Unified Modeling Language (UML)

UML is developed by the **Object Management Group** (http://omg.org) as a standard language for software systems development and analysis.
It gives possibility to represent the system architecture and functioning dynamics on the base of different types of diagrams, created using standardized graphical notations.

In the UML version 2.2 there are 14 types of diagrams divided into 2 categories:

Structure diagrams (such as Class Diagram, Object Diagram, Deployment Diagram, Package Diagram)

Behavior diagrams (such as Sequence Diagram, Activity Diagram, Use Case Diagram, Timing Diagram).

UML Sequence diagrams

Sequence diagram is a particular case of UML interaction diagrams representing (or modeling) the dynamic behavior of a system through the tracing of the sequence of interactions that occur between the objects or components of a system. The Sequence diagram is one of the most widely used dynamic diagrams in UML.

For creating the Sequence diagrams for AliEn functionality, I have used **UML::Sequence** tool. It takes as input tool-specific textual description of the code and produces the Sequence diagrams.

To my knowledge, there is no open source software tool that allows creation of Sequence diagrams directly from Perl-written code.

How the diagrams are created

The creation of diagrams for the AliEn services has been done by:

0. Installing the **UML::Sequence** tool;

1. Manually converting the Perl code to the text format (plain text) file;

2. Passing the text file to the **UML::Sequence** modeling tool.

0. Installing the UML::Sequence tool

1. Enter to the Perl CPAN shell (it is assumed that the Perl is already installed)
#> perl -MCPAN -e shell

2. Install the GD::Text::Wrap module which is prerequisite for UML::Sequence
cpan> install GD::Text::Wrap

3. Install the UML::Sequence module from CPAN shell
cpan> install UML::Sequence
This will require and ask for installation of the following dependencies:
XML::DOM,
GD,
XML::Parser::PerlSAX,
XML::RegExp

1. Converting the Perl code to the text format

A snippet of Perl code (PackMan.pm)

```perl
sub definePackage
{
 ...
 my $lfnDir=lc($self->{CATALOGUE}->{CATALOG}->
GetHomeDirectory()."/packages");
 ...
 while (my $arg=shift){
   if ($arg=~ /^-?-se$/ )    { ... }
   ...
 }                                          check arguments
 ...
 $self->{CATALOGUE}->{CATALOG}->isFile($lfn) and
$self->info( "The package $lfn already exists") and return;
  ...
}
```

Corresponding plain text (input.txt)

```
AliEn::PackMan::
AliEn::PackMan.definePackage
  AliEn::Catalogue.GetHomeDirectory -> $lfnDir
  AliEn::Catalogue.check arguments
  AliEn::Catalogue.isFile
  ...
```

2. Passing created file to the UML::Sequence tool

The plain text (input.txt)

```
AliEn::PackMan::
AliEn::PackMan.definePackage
  AliEn::Catalogue.GetHomeDirectory -> $lfnDir
  AliEn::Catalogue.check arguments
  AliEn::Catalogue.isFile # checks that the package $lfn ...
  AliEn::UI::Catalogue.execute &#40;mkdir -p&#41; -> done
  AliEn::UI::Catalogue.execute &#40;addTag&#41; -> done
  AliEn::UI::Catalogue.execute &#40;add&#41; -> done
  AliEn::UI::Catalogue.execute &#40;addTagValue&#41; -> done
  AliEn::Database.do &#40;update&#41; # Package $lfn added!!!
```

1

The diagram generation command(s)

2

$> *genericseq.pl* **UML::Sequence::SimpleSeq input.txt \
|** *seq2rast.pl* **> pack-define.png**

or

$> *genericseq.pl* **UML::Sequence::SimpleSeq input.txt \
|** *seq2svg.pl* **> pack-define.svg**

The generated diagram
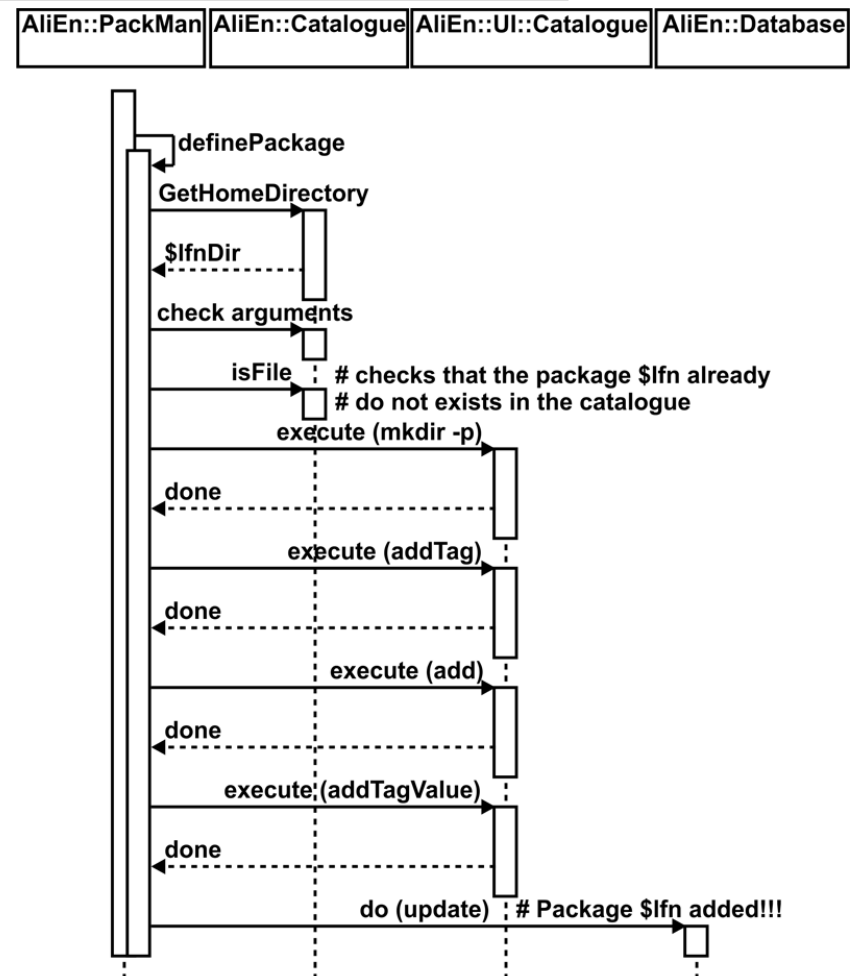
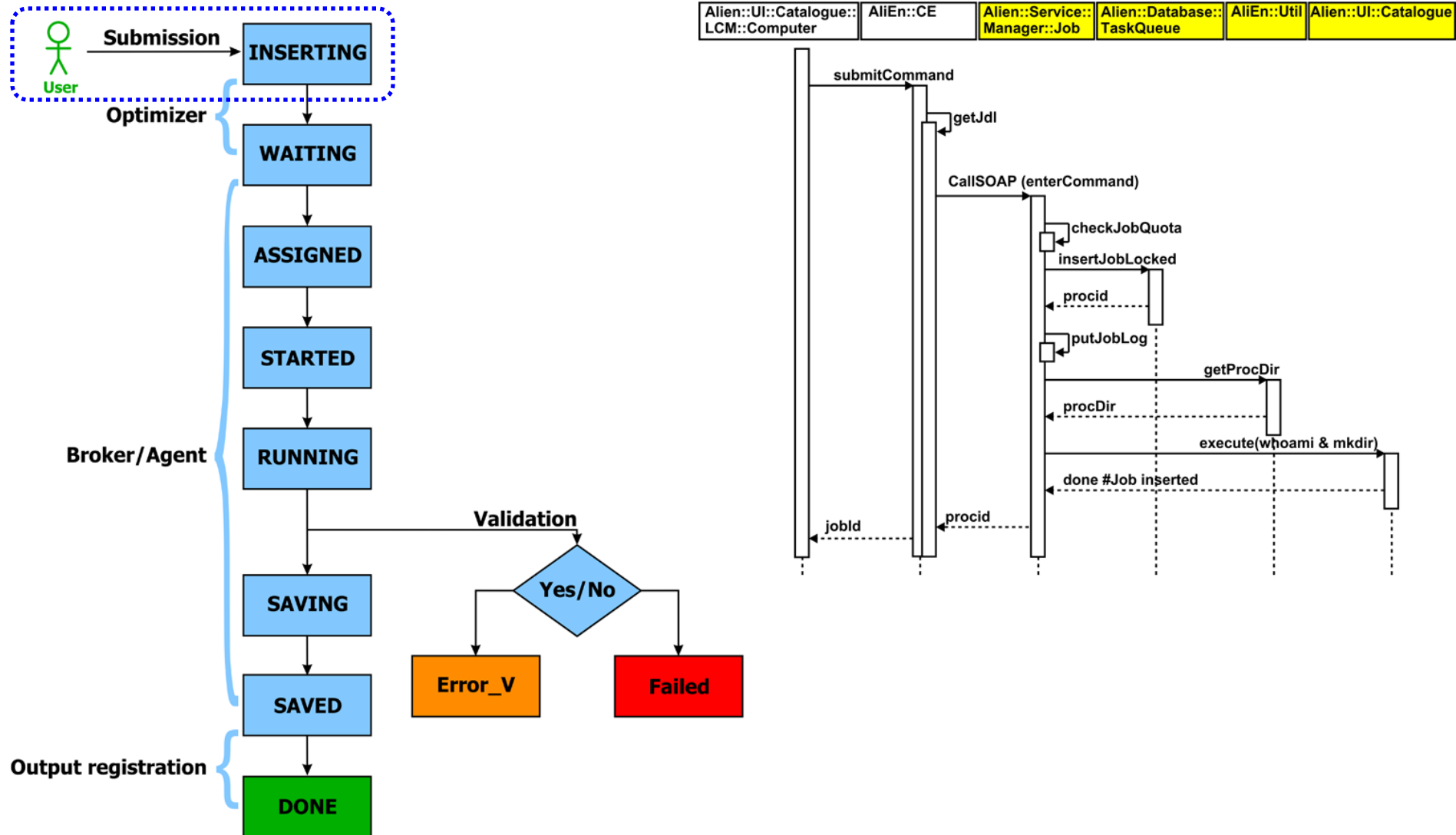# How to read the generated diagram
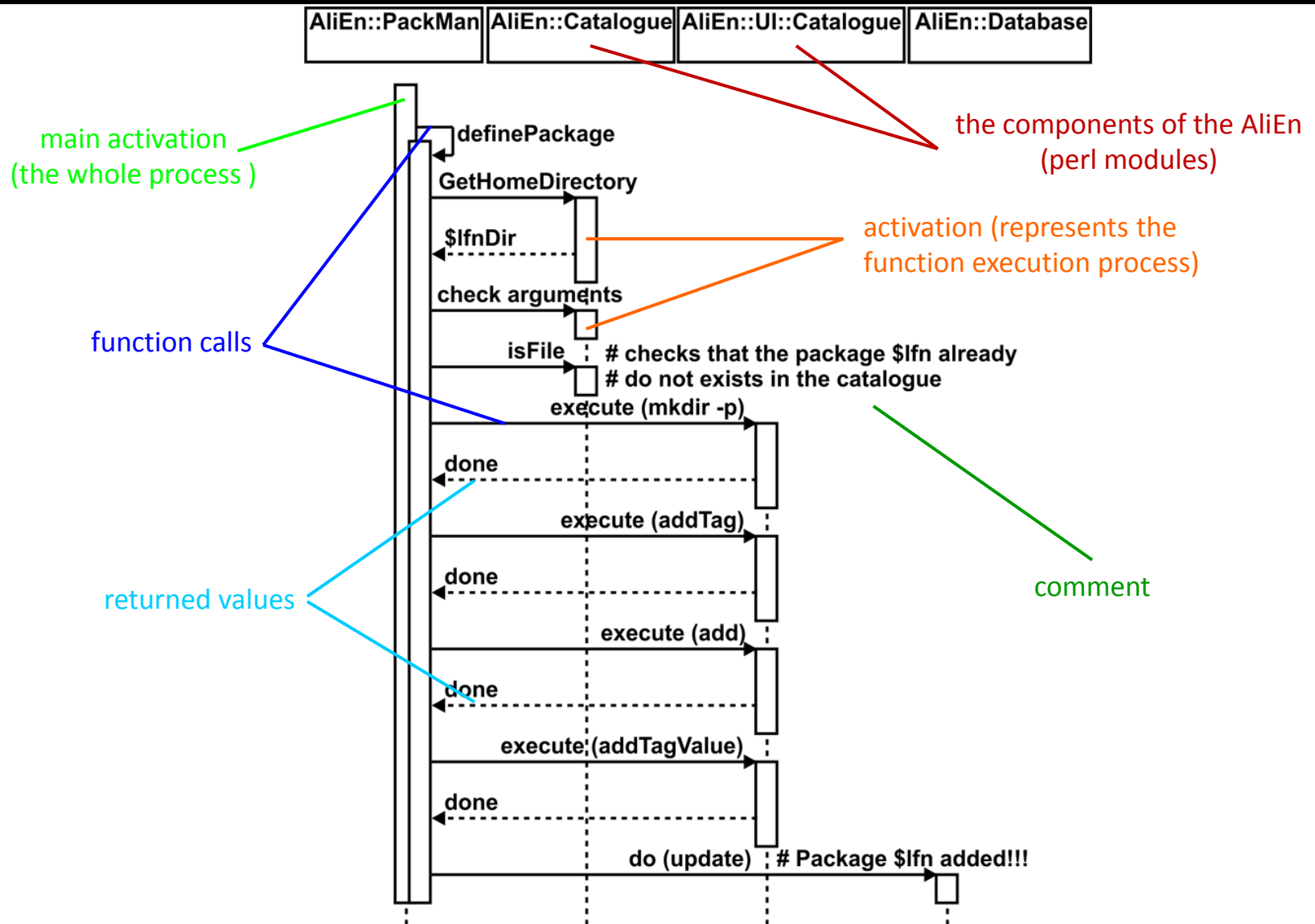
Sequence diagrams for AliEn job execution part
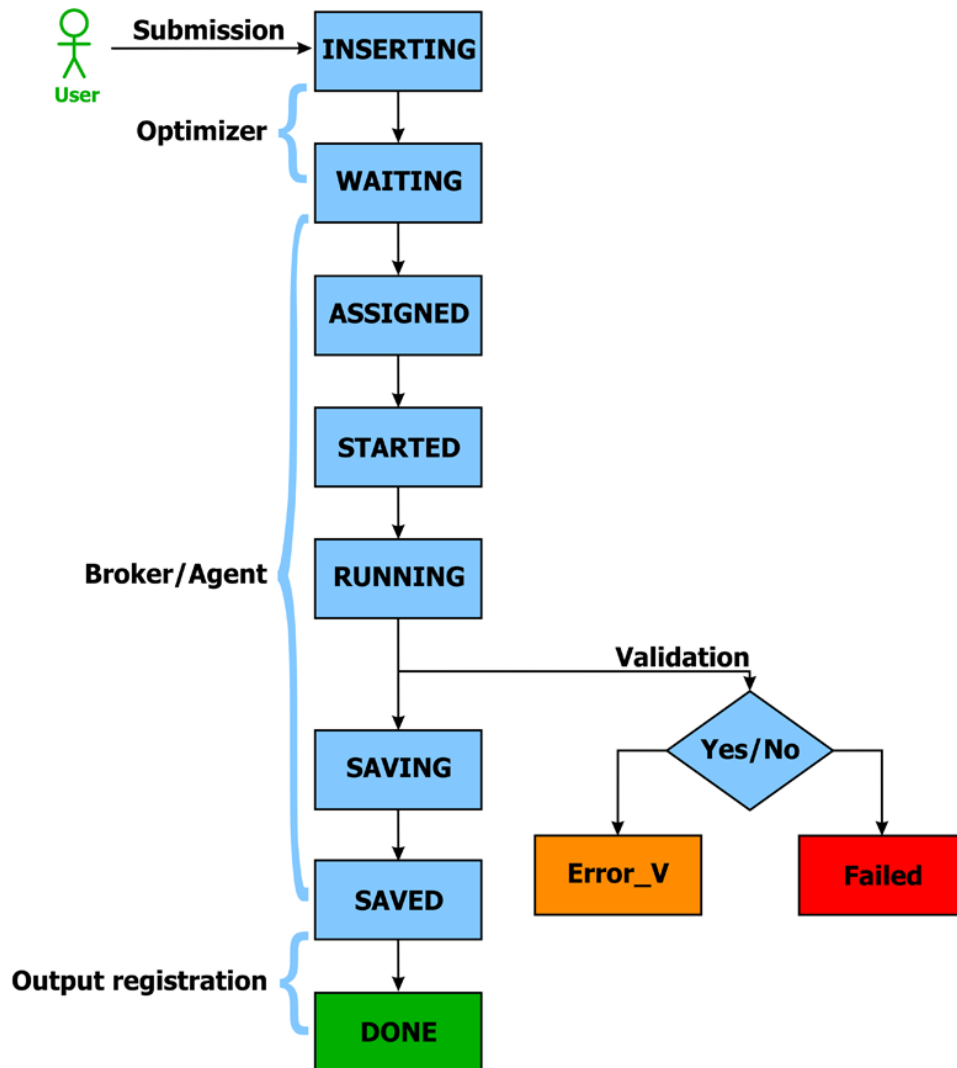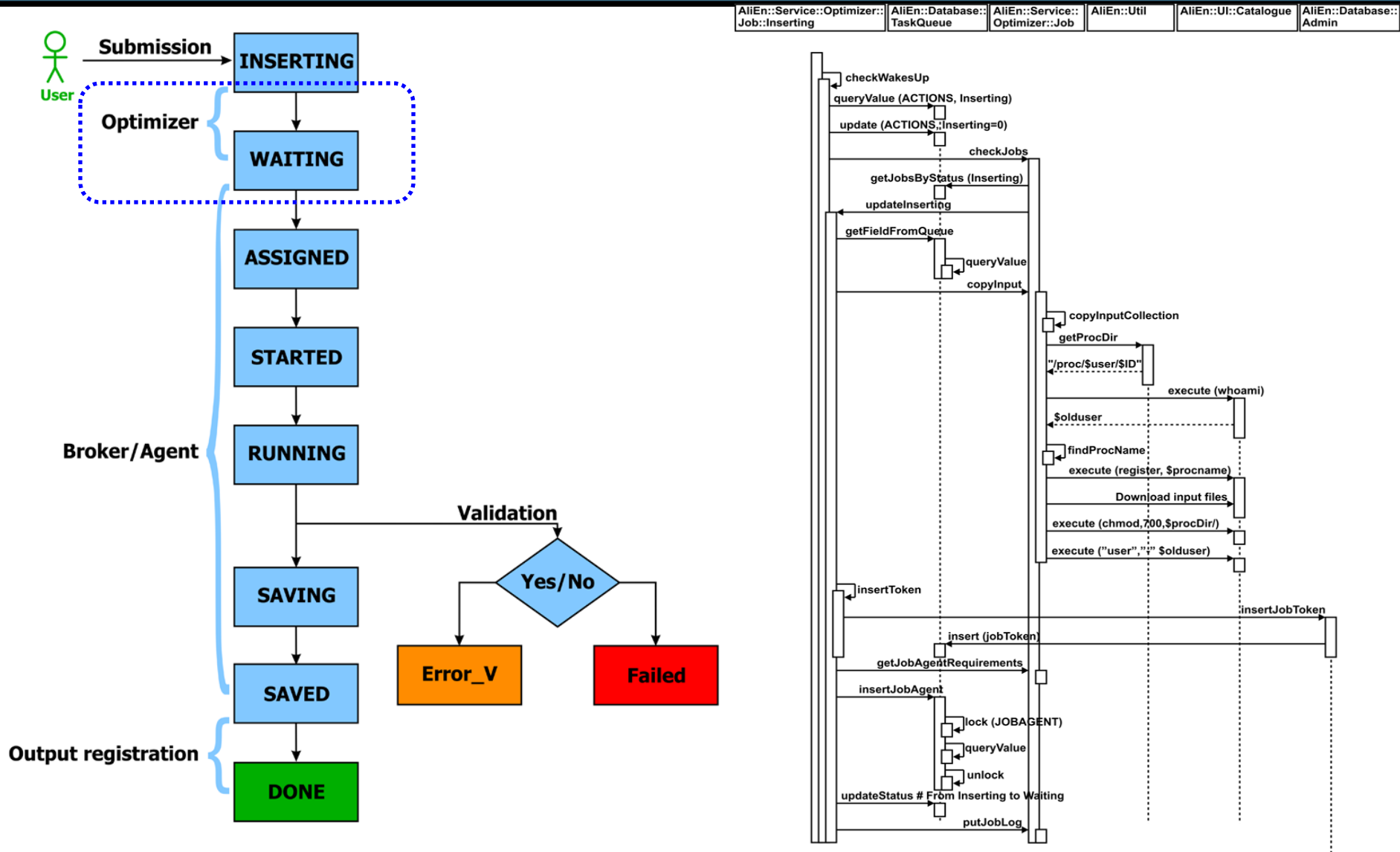

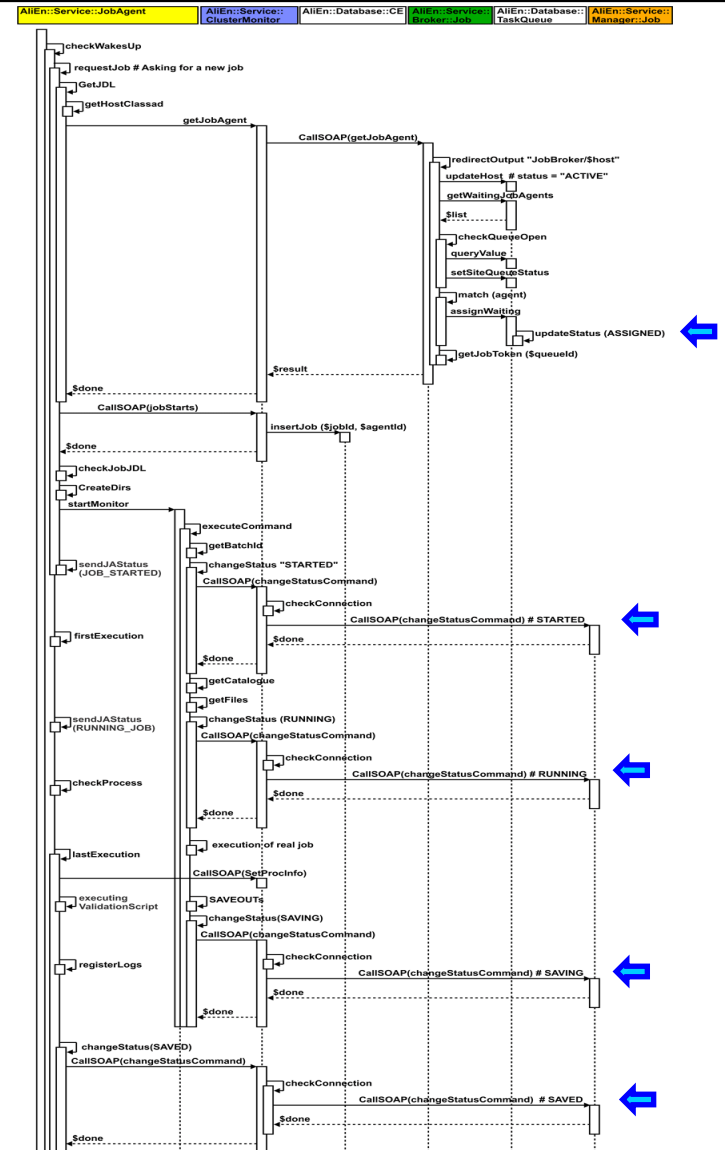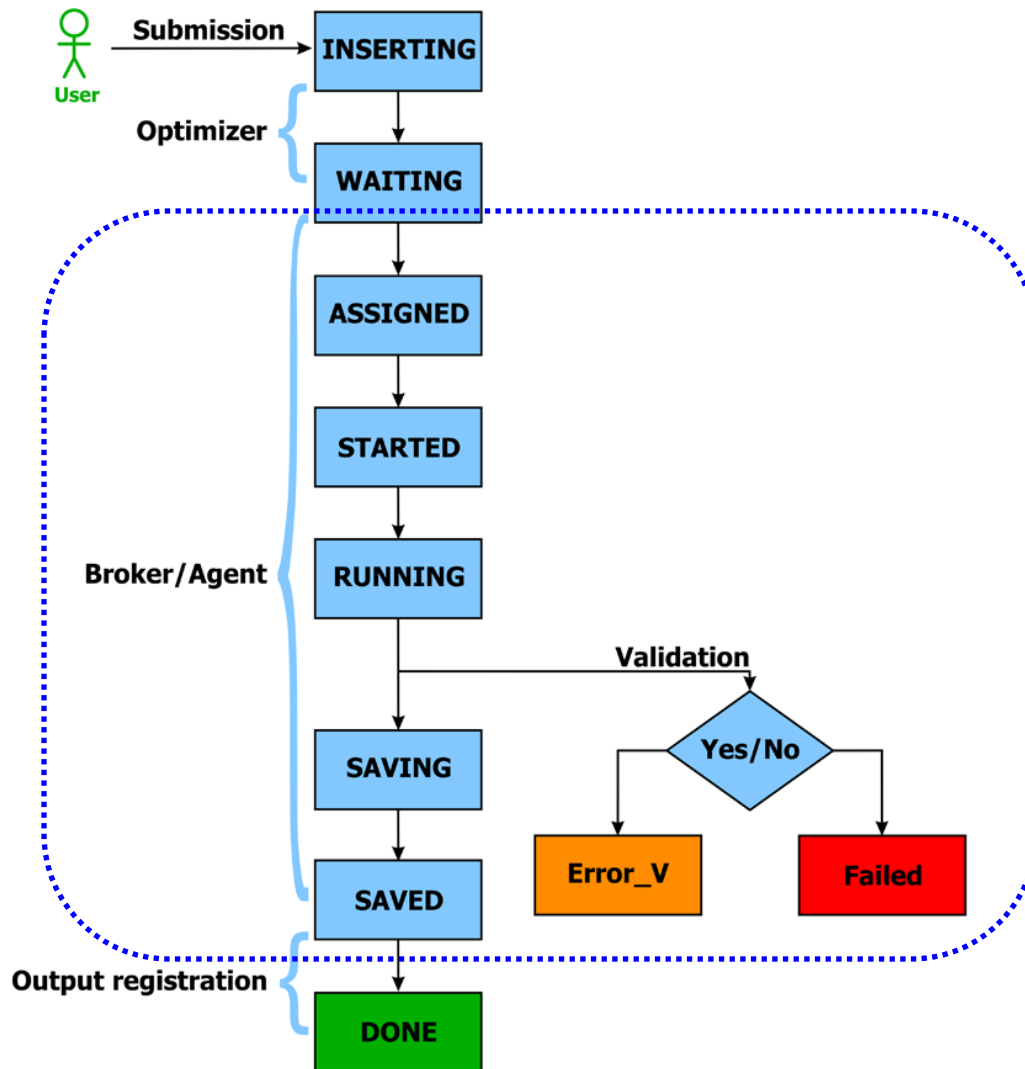The whole chain of successful statuses the job execution, after its submission by a user.

Submission stage

Optimizer stage

Broker/Agent stage

Output registration stage

Sequence diagrams for AliEn PackMan service

AliEn provides a service for automatic installation (on demand of Job Agent or upon an explicit request by the software manager of ALICE Virtual Organization) of the software packages required by the job. The installation, upgrade, configuration and removal of application software packages are done by the AliEn Package Manager (PackMan) service.

The basic commands of the PackMan service are the following:

1. **packman define** - After execution of this command, the software to be installed is registered in the AliEn File Catalogue;
2. **packman list** – The list of all the packages that are defined in the system is returned;
3. **packman listInstalled** – To return the list of all packages installed on the site;
4. **packman install** - By this command PackMan installs a registered in the AliEn File Catalogue package (and all its dependencies) in the package repository of the PackMan.
5. **packman undefine** – To remove a package from the AliEn File Catalogue.

http://alien2.cern.ch/index.php?option=com_content&view=article&id=103%3Apackman&catid=7&Itemid=64

Conclusion

In order to support the effective involvement of new developers in the **AliEn** development process, the ALICE Offline team has decided to develop a whole set of the UML diagrams describing the dynamic behaviour of AliEn services.

By now the diagrams describing the AliEn **Job execution** part and **PackMan** service have been created and presented on the AliEn official documentation web pages for the developers.

The work on the development of the diagrams for the other AliEn services is under way.

Acknowledgments

I am grateful to **Latchezar Betev** for suggesting me this work and making numerous useful comments.

I appreciate the collaboration with **Pablo Saiz** in the development of presented UML diagrams.

The discussions with the colleagues from ANSL and CERN: **Federico Carminati, Ara Grigoryan**, **Artem Harutyunyan**, **Arsen Hayrapetyan**, **Narine Manukyan**, **Vardanush Papikyan** and **Lola Saiz Santos**, helped me to understand many details of AliEn and UML.

Thank you