

Rules concerning the extraction and the restitution of the sources of Code_Aster

Summary:

One gives here the rules which the developers of Aster must comply with when they recover a source official (extraction) and when they propose a modification of a source official (restitution). These rules clarify points also approached in the expansion plan [A2.01].

Contents

1 Introduction.....	3
2 Rules of extraction.....	3
3 Rules of restitution.....	4

1 Introduction

One explained in [D1.01.01] how a developer could “overload” Code_Aster with sources staffs. The sources about which one speaks here are:

- routines FORTRAN, or C
- Python sources
- catalogues,
- CAS-tests.

A classical development of Aster results in general in:

- modification of certain sources,
- the writing of new sources.

The modification of sources supposes their preliminary extraction (recopy) sources official. The paragraph §2 relate to the rules of extraction.

The finished development, it can be “restored”: one puts up to date the sources then official. The paragraph §3 relate to the rules of restitution.

2 Rules of extraction

R1	The extraction of a source is the recopy of the official version of this source. (To avoid the copies of copies).
R2	When a source must be modified in the sight of a future restitution, “to note” this source [D1.02.01] (order asno of astk). This notation indicates to the other developers the intention to modify this source and makes it possible to detect possible futures conflicts of notation.
R3	When a source must be modified and restored: not to touch the “line” MODIFICATION which contains the date of the last modification of the source. This date makes it possible to check during the restitution that the source of origin (before modification) is well last the source official. This prevents that a modification “crushes” a preceding modification. This rule is capital: an intentional violation would be very badly interpreted... and sanctioned.
R4	Not to modify (for the pleasure or by taste) the presentation of the source: the indentation, the name of the variables,... except if is to make it in conformity with the rules presentation or of programming [D2.02]; because this artificially increases the volume of the “delta of evolution” which one preserves: the relevant modifications do not jump any more to the eyes.
R5	If the source extracted were already noted by another developer: <ul style="list-style-type: none">• to contact to make him confirm its intention to restore this source,• to act in concert with him for the order of the restitutions,• if it will restore before you, not to launch out head lowered in a modification of great width of this source, because it will be necessary to start again the modifications in the source official resulting from its evolution (cf R3).
R6	Not to note sources unnecessarily: you can obstruct the other developers (cf R5). In particular, to note a source only one intends to restore in 6 months!
R7	When one noted a source by error, to indicate it [D1.02.01] (order asdeno).

3 Rules of restitution

R1	To use the automatic mechanism of restitution of <code>astk/agla</code> : order <code>asrest</code> [D1.02.01]. This mechanism imposes that a restitution is autonomous: the official code modified by the restitution must be able to carry out the CAS-tests of nonregression.
R2	A "asrest" erases another of them: there is no pile of restitution for a given developer. A developer cannot thus make more asrest per week for a given version of code (NEW10 or NEW9).
R3	A "restitution" (sources modified + sources added) will not be effective (modification of the sources official) only after the procedure of update of the code made by the administrator [D1.02.02 §1.2]. This action has leu in general every Wednesday morning.
R4	To destroy the sources of a restitution only after having received (and read) the "e-mail" announcing the update of the code.
R5	Any restitution must be accompanied by a file of history (standard <code>histor</code> of <code>astk</code>) in which the developer must list the cards of REX that it "corrects". Each one of these cards must be last in the "solved" state. While passing the card in the "solved" state, it is necessary to add the last "post" in the REX who summarizes the card: request/problem, corrections carried out,... It is the latter "post" which will be examined in EDA every Monday morning.
R6	To carry out the "pre-eda" ordering of <code>Astk</code> before the Monday the 9th H. "to present" its restitution in meeting of EDA Monday. To bring a transparency of its history file.
R7	To think of the validation of its restitution: addition/modification of CAS-tests.
R8	To destroy a source (or to change it library) to see [D1.02.02 <code>unigest</code>].
R9	When a new unit of source is added. This unit must contain a line 'ADDITION' whose syntax depends on the type of the unit [D1.02.02].