XMM-Newton

XMM-Newton Science Analysis System 16.0 scientific validation

XMM-SOC-USR-TN-0028 Issue 1.0

C.Gabriel, I. de la Calle, A.Ibarra, R.González-Riestra,
E.Ojero, J.V.Perea, P.Rodríguez, R.Saxton, M. Smith, M.Stuhlinger, A.Talavera
XMM-Newton Science Operations Centre

January 18, 2017

Revision history

Revision number	Date	Revision author	Comments
0.9	December 22, 2016	C. Gabriel	first draft
1.0	January 18, 2017	C. Gabriel	after revision & comments from co-authors

Contents

1 Introduction

The SAS scientific validation is performed on a standard set of XMM-Newton observations, which cover all commissioned observational modes, and a number of observations, specially chosen for testing new / special aspects of the data reduction corresponding to the version to be validated. Table 1 lists all the datasets used for the validation of SAS version 16.0. Some of these observations are particularly suitable to test calibration-related items, as specified in the rightmost column of Tab. 1. These datasets are partly intended as a standard reference, which has been and will be used to verify the performances of all SAS versions. However, additional datasets may occasionally be used to test version-specific SAS items. This is the case, for instance, for the datasets discussed in Sect. 2 of this report. Datasets discussed in a given report and not listed in Tab. 1 do not belong to the reference datasets, and are therefore not intended to be discussed in later SAS versions validation reports.

1.1 Methodology

The SV for SAS v16.0 consisted of the following steps:

1. all the datasets listed in Tab. 1 were processed through the SAS 16.0 based testing Pipeline System (PPS) running at the SOC, and

2. the same datasets were also processed through the SAS reduction meta-tasks: `e[mp]proc`, `om[ifg]chain`, `rgsproc`

3. all the SAS threads were ran as documented, for checking the integrity of the software and the validity of the threads

4. products generated by the above steps were used as basis for the *interactive SV analysis*. Standard scientific products (images, light curves, spectra, source lists) were generated and analysed. This allowed us to:

   - test the SAS interactive tasks.
   - verify the calibration accuracy obtained with SAS v16.0, and compare it with the expected accuracy on the basis of the calibration status at the time the SV is performed.

5. in addition the whole cross-calibration database has been reduced by standard analysis scripts based on SAS but including also model fitting through Xspec.

Table 1: SV datasets

Instrument	**Mode**	**Object**	**Revolution Obs. ID**	**ID**	**Calibration item**
EPIC MOS	Full Frame	Lockman Hole	544 0147511601	1	Astrometry + source detection
	”	G21.5.09	060 0122700101	2	Effective area
	”	M31	2847 0761970101	2	Extended source
	Small Window (W2)	Mkn 421	165 0099280201	3	
	Large Window (W3)	PKS0558-504	153 0129360201	4	Effective area
	Timing Uncompressed	Her X-1	207 0134120101	5	Timing
EPIC-pn	Full Frame	Lockman Hole	544 0147511601	1	Astrometry
	Full Frame/Small Window	PKS0558-504	153 0129360201	4	Effective area
	Full Frame	M31	2847 0761970101	2	Extended source
	Large Window	AB Dor	185 0133120201	6	
	Small Window	PKS0558-504	084 0125110101	7	Effective area
	Fast Timing	Her X-1	207 0134120101	5	Timing
		Crab	698 0160960201	8	
	Fast Burst	Crab	411 0153750301	9	Timing
		Crab	411 0153750501	10	Timing
	Extended Full Frame	G21.5-0.9	060 0122700101	2	Effective area
	Slew Data		1388 9138800002	18	Slew data processing
	Slew Data		1450 9145000003	19	Slew data processing
RGS	SPEC+Q	PKS0558-504	084 0125110101	7	
”		Mkn 421	165 0099280201	3	Effective area
	”	AB Dor	185 0133120201	6	Wavelength scale
	”	AB Dor	338 0134521301	11	Wavelength scale
	”	AB Dor	462 0134521601	12	Wavelength scale
	”	AB Dor	572 0134522201	13	Wavelength scale
OM	Image Mode	BPM 16274	261 0125320701	14	Photometry
	Fast Mode	X1822-371	228 0111230101	15	
	FF Low Resolution	BPM 16274	261 0125320701	14	Astrometry
	Optical grism	Hz2	503 0125910901	16	Wavelength scale & flux calibration
	UV Grism	HD13499 (offset)	657 0125911301	17	Wavelength scale & flux calibration

1.2 Calibration data to be used

The calibration data to be used for this version was derived from the full public calibration constituents as of 15 November 2015, plus the following components which at this date were not yet public:

- EPN_FWC_0001.CCF (made public with the SAS 16 release, since it is related to a new task, `evqpb`),
- RGS1_EFFAREACORR_0011.CCF, RGS2_EFFAREACORR_0011.CCF (made also public with the SAS 16 release, since they are related to a new non-default effective area correction).

2 New and updated in SAS 16.0

V. 16.0 is a main yearly release of the SAS, containing some new capabilities of the package. The main item of the upgrade, though, is the migration to new compilers. Of upmost importance is the migration of the Fortran compiler used, since from this version on building of the SAS should not require anymore the NAG compiler, but will be based on the use of the GNU gfortran compiler. This radical change should not only imply savings, but also help in the future external institutions to compile and build SAS, or derivatives from it. It has to be noticed that, together with the migration to gfortran a higher version of the GNU compilers (GNU C/C++ and the corresponding gfortran compilers) had to be used, since the version used before (4.8.x) was not able to cope with some functions included in SAS tasks. The last public version has been chosen at the end for SAS 16 (GNU C/C++ 6.2, released in August 2016). A good discussion on the different alternatives with diverse C++ and gfortran compiler versions can be found in the XMM-Newton technical note Ojero 2016.
Many tasks were affected by this change, due to significant differences in both compilers' standards. Especially the interfaces between C++ and Fortran inside the SAS had to be modified. We wanted to keep the ability to compile SAS with NAG, at least for a reasonable time. Some special mechanisms had to be introduced for maintaining that ability, with a discrimination in the source code. This makes possible that we produce for this release a gfortran and a NAG compiled version, the latter just for testing purposes.
The large number of changes introduced in some 80 tasks, but especially in interface tasks, let us expect some numerical deviations between the results obtained in the different compilation tracks. Therefore the validation of the gfortran track should be obtained by qualitative comparison. However, first tests on a reduced data set have shown minimal differences. These tests should be extended and if the results are as positive as found already they could represent a fundamental part of the validation and reduce the demands on the qualitative comparison between NAG and gfortran compiled versions.

2.1 Updated in SAS 16.0: change of a large number of tasks for being able to use the gfortran instead of the NAG compiler

Every single task has some level of harness testing at the building stage, which has to be passed for acceptance into the new version built. A global comparison can be only done qualitatively, if numerical differences are found between products obtained with the gfortran and NAG fortran compiled versions. For that purpose, all the standard data should be reduced, so as to test the software related to all the different observation modes, in the two incarnations produced (gfortran and NAG). Final products should be compared according to the scheme given in Tab. 2, referring to all XMM-Newton instruments.

2.2 New in SAS 16 - `evqpb`, a task for providing for any given EPIC science exposure an associated Quiescent Particle Background (QPB) events file

The task extracts for a given EPIC exposure a qpb events file using the filter wheel close data repository. The produced qpb events file can be used for correcting spectra and images for this background component. Since this is only possible for Full Frame mode in the case of MOS and restricted in the first implementation to this mode also in the case of PN we should use for validation a suitable extended source observation in Full Frame. We have added therefore to the standard set, for future reference, an observation of the galaxy M31 (0761970101), off-centre but well covered by both MOS1 and MOS2.

2.3 Upgraded in SAS 16.0: `ebkgreg`, a task for determining optimal background position for a source in the FOV - now also for EPIC-MOS

This task had been originally implemented only for PN. Now it has been extended for working with MOS data. Interactive use should be tested, including its behaviour by crowded images. A number of observations with many sources in the FoV (e.g. deep observations of the Lockman Hole area) as well as with extended sources (eg. clusters) should be used for the validation.

2.4 New in SAS 16.0: `edetect_stack`, a new meta-task for source detection by overlapping observations

This meta-task performs standardised EPIC source detection on overlapping fields of observations, taken at different epochs or and in Mosaic Mode. Still at experimental level, it should constitute in a near future the basis for the production of a catalogue of stacked observations. While extensive validation will be only in the future possible, testing with a limited number of datasets should be part of this validation.

2.5 New in SAS 16 - new effective area correction factors per instrument and order

Increasing evidence has been observed for time dependent systematic changes in the ratio of fluxes per wavelength period between RGS1 and RGS2, in both orders. Kaastra et al. (2011)

Table 2: Comparison elements

Calibration Item	**Test products**	**Test items**
Astrometry + source detection	Source lists	Nr. of sources, positions, fluxes
Effective Area	Spectra + Light curves	Model results, harness ratios
Timing	Light curves + Fourier transforms	Periodicities
Wavelength scale	Spectra + Light curves	Line positions and strengths
Photometry	Source lists	Fluxes

have shown the presence of "wiggles" in the effective area, proposing a new empiric correction. This has been implemented within `rgsrmfgen` as a non-default correction (parameter *witheffectiveareacorrection*), extracting the values from new extensions in the RGS[1][2]_EFFAREACORR CCF files (issue 11 of both files is the first one implementing this).
Validation of the new correction will be done together with the validation of the CCF files involved.

2.6 New in SAS 16 - use of `embadpixfind` for RGS bad pixel / column finding

The task `embadpixfind` has been adapted for its use on the RGS data reduction chain. The corresponding (default) task `rgsbadpix` used for flagging bad pixels and columns has been reported to discard in certain circumstances columns belonging to bright emission lines, when actually these columns should be perfectly valid.
A thorough validation has to show the advantages in those cases, but also the results of applying this new bad pixel recognition by continuum sources, i.e. in the absence of emission or absorption lines.
Validation should therefore be done comparing the results of tens of observations showing emission lines at different wavelengths (ie. AB Dor, HR1099) using both methods for flagging bad columns, as well as of continuum observations at different signal levels (ie. different observations of Markarian 421). Please notice that the validation of this task has failed for SAS 15.0. Dedicated validation has been performed for SAS 16. The results are exposed in section 4.7.

3 Validation results

3.1 Validation schedule

This SAS version should be released in diverse Linux OS 64 versions as well as two MacOs versions, to cover a broad band of kernels. According to the collected download statistics we perform with every release of SAS, the last version, SAS 15, has been used by less than 5% of all SAS users in its 32 bit versions. Moreover, it has already been announced in XMM-Newton Newsletters and to the Users Group -e.g. Newsletters #158, in December 2013, and #180 in May 2016 and UG meetings in 2014 and 2016- that 32-bit binary versions would be discontinued. The UG has agreed in 'eventually withdraw the 32 bit version'. Therefore we see as justified the production of just one Virtual Machine for 32 bit OSs, with a dedicated limited run for qualitative validation, once the bulk of the validation has been performed, but not to distribute it, and handle it to users only on specific demand.
The schedule for the validation foresees a total of around 6 weeks for performing the different tasks (for the period from going into release track mode to final release). This is the projected schedule with the different milestones:

- SAS into release track mode - 7 October 2016
- SAS builds on different platforms - 10 October
- SAS 16 binaries (at least 1 platform) - 10 October
- Processing of all the standard datasets (in both gfortran and NAG fortran compiler versions) - finished - 12 October
- Installation of SAS 16 binary in XCal grid - 13 October
- Communication to validators about success and data location - 13 October
- Cross-comparison of gfortran vs NAG results - 24 October
- Preparation of a SAS 16 based PPS test version - 25 October
- Processing of standard datasets by testing pipeline - 28 October
- Processing of standard datasets by all binaries + cross-checks - 28 October
- Processing of XCal archive - 28 October
- First I/A analysis of standard data - to be ready by 4 November
- Integrity checks running all the existing SAS threads - to be ready by 4 November
- Evaluation of XCal - to be ready by 4 November
- Screening of PPS processed standard datasets - 4 November
- Dedicated analysis - to be ready by 4 November:
    1. `evqpb` tests
    2. `ebkgreg` tests
    3. RGS new bad pixel finding method

- Summary reports - 11 November
- SAS VM produced - 11 November
- Processing of all standard sets with VM - 11 November
- Final SV individual reports due on 18 November
- Release notes + SAS 16 web pages contents ready + XMM Newsletter text - 18 November
- SAS 16 distribution tar files ready - 18 November
- SAS 16 release - 23 November
- Final SV Report compilation - 23 November

While the first part of the validation plan could be followed as planned, the difficulties found when comparing gfortran vs NAG Fortran compiler version results produced serious delays. Subtle problems have been found behind small deviations, which could be only solved with large efforts. Especially the fact that new problems appear with every new platform used in the transition to gfortran forced us to many more iterations than planned. The result was a 3-4 weeks delay, which combined with the Christmas holidays, moved the release to early 2017.
Very early in the process of preparation of the new version, it was recognised that the transition to using gfortran had to be combined (for compatibility reasons) with a higher version of the GNU C/C++ compiler. SAS 16 was produced using version 6.2. Using this very recent version of the C++ and gfortran compilers, we had to follow the most recent standards for C++ and Fortran. Many tasks had to be adapted to newer standards. Since one of the main elements of this validation should be the check that compilation with gfortran was not introducing artifacts, the best way was direct comparison with data processed with a binary obtained with the NAGfor compiler. This forced us to maintain the full compatibility with NAGfor.

3.2 Processing of standard datasets

After several iterations, all the data listed in Tab. 1 have been processed with the full data reduction scripts used for validation without any failure, on the main testing platform on CentOS6.8 (free twin of RHEL6.8). The whole testing was also performed with the NAG fortran binary version for comparison. Furthermore, reduced testing with all four distributed versions (2 Linux and 2 MacOS) was also performed.
The difficulties posed by the migration to gfortran, were numerous and affected a large number of tasks. It follows the list of software changes which were necessary for obtaining a perfect building package:

- New definition of array descriptors (dope vectors) - In Fortran 95/90, pointer arrays are stored in memory with some details of how the array is organised in memory. When this array pointers are passed to other language (i.e. C++), these details or "meta information" have to be "mapped" on the C++ side, so that the other language correctly interprets the descriptors to obtain the information it needs. Array descriptors definition changes from one compiler to other.
- Memory mapping of array descriptors - For using gfortran we had to explicitly store a matrix contiguously in memory.

- Usage of specific MACRO statements for NAG for modules - In few cases, we had to include macro preprocessor statements to keep backward compatibility with NAGfor. This was strictly necessary because at the end of the validation "gfortran-derived" products should be compared with "NAGfor-derived" ones.
- Definition of new naming convention for Fortran 90/95 precompiled modules - NAG fortran and gfortran follow different standard regarding modules naming conventions.
- Usage of intrinsic functions - There are few cases where the use of, for example the "real" intrinsic function to convert a number into a real, depends on the compiler.
- Other examples are: get_environment variable, Leading blanks spaces removed properly.
- Implied loop - NAGFor standards differs from the gfortran standard regarding the manipulation of arrays in loops.
- Usage of reserved words - gfortran is more strict in the usage of reserved words.
- Integer to String conversion - gfortran is more strict with the conversion of integers to strings.
- Namespace errors - gfortran is more strict regarding the scope of modules names. Some modules used in several packages had to be defined as "public"
- Memory allocation - Usage of an optional parameter as part of an allocatable variable definition is not possible in gfortran.
- Array initialisation - Non initialised arrays in gfortran are undefined. This may cause sudden segmentation faults.
- Runtime errors - gfortran is more strict with the definition of parameters in subroutines. We had to declare some parameters as "inout" instead of "out", otherwise a seg-fault was thrown out at run-time.
- Overflow computations - gfortran is more restrictive regarding the size of a variable and therefore variables have to be carefully defined in their sizes. We had to recast some variables to integer kind 9, for example, to avoid multiplication overflow in some calculations.
- Pointers - In gfortran all pointers that are passed to C++ have to be initialised to null. Not intitialising the pointers produce seg faults following at run-time.
- NaN treatment - gfortran does not handle NaNs in the same way as NAGFor. Therefore, a check if the content of a variable is a valid number is necessary instead of checking an overflow in the results.

3.3 Comparison of products from the two main binaries, compiled with NAGfor and gfortran

Once the problems posed by the migration to a compilation with gfortran were solved, the comparison between the products from the binaries produced using the different compilers was surprisingly simple. The level of agreement between the products was by far exceeding our most optimistic expectations. The products were almost identical, few minor differences have been found, which are most probably originating in randomisation effects in certain areas. By none of the EPIC and RGS calibrated event files the difference in events went over one event out of typically $10^6 - 10^7$, in most cases there were no differences at all. Of course derived spectra, light curves and images were therefore practically identical.

3.3.1 EPIC data - standard set

Comparison of the EPIC data products obtained with the SAS versions compiled with NAGfor V5.2 + GNU C/C++ V4.8.5 and GNU C/C++ V6.2.0 including gfortran:
Only few differences in some of the columns in the PN event files were found. The largest difference was found in an event file (obs. 0153_0129360201) with 20 events showing a difference of 1eV in a total of more than 1.5E06 events. This can be attributed to small numerical differences when calculating / correcting the events' energies.
In another PN event file (corresponding to the observation 0084_0125110101) 4 events (out of 5.E05) show different PAT_SEQ and PATTERN values. This, although absolutely not worrying (and not considered a showstopper), is under investigation.
No further differences found.

3.3.2 RGS data - standard dataset

Output products of the whole processing by `rgsproc` have been compared with the results of processing with previous versions of SAS. No significative differences have been found. Detailed studies are referred further below.
Details about the comparison of the RGS data products obtained with the SAS versions compiled with NAGfor V5.2 + GNU C/C++ V4.8.5 and GNU C/C++ V6.2.0 including gfortran:
- Merged event lists (*merged*, 35 files)
Differences found in:
P0125110101R1S019merged0000.FIT: 1/505418 event, 1.5E-11 rad in XDSP_CORR
P0761970101R1S004merged0000.FIT: 1/225614 event, 1.5E-11 rad in XDSP_CORR
- Filtered event lists (*EVENLI*, 35 files)
Differences found in:
P0125110101R1S019EVENLI0000.FIT: 1/209301 event, 1.5E-11 rad in XDSP_CORR
- Source lists (*SRCLI*, 35 files)
Differences found in:
P0099280201R2S002SRCLI_0000.FIT: 1.5E-05 deg in FOV_PHI
P0133120201R2S002SRCLI_0000.FIT: 7.6E-06 deg in FOV_PHI
P0134521601R1S001SRCLI_0000.FIT: 7.6E-06 deg in FOV_PHI
P0153750501R1S004SRCLI_0000.FIT: 7.6E-06 deg in FOV_PHI
P0160960201R1S004SRCLI_0000.FIT: 1.5E-05 deg in FOV_PHI

- Extracted count spectra (*SRSPEC*, 70 files)
No differences
- Extracted background spectra (*BGSPEC*, 70 files)
No differences

3.3.3 SAS OM data processing

Some small differences have been found between data processed with the SAS versions compiled with NAGfor V5.2 + GNU C/C++ V4.8.5 and GNU C/C++ V6.2.0 including gfortran:

- Keywords in headers of fits files extensions - Some keywords with photometric values computed by SAS, e.g. magnitude limits, may show small differences (in the 4th decimal place)
- Source lists show different number of sources in source lists - combined source lists may have different number of detected sources.
  old: P0125320701OMCOMBOBSMER0000.FIT 359 sources
  new: P0125320701OMCOMBOBSMER0000.FIT 356 sources

  old: P0111230101OMCOMBOBSMER0000.FIT 6413 sources
  new: P0111230101OMCOMBOBSMER0000.FIT 6411 sources
- Different photometry values - P0125910901OMS005SWSRLI0000.FIT one source is considered in the "new" system as extended and therefore its photometry is completely different.
- Characterisation of the source - For many sources there are small differences in the parameters FWHM and PA (mostly minor differences)

3.3.4 Test pipeline checks

Two test pipeline versions, built on the basis of the gfortran and the NAGfor SAS 16 binaries, have also been used for the data reduction of the standard sets and the posterior comparison of the products from the one and the other. No errors were found in neither version, and the outputs were correctly derived. Comparisons of the products have shown only few minor differences in the number of detected (faint) sources, which can be attributed to numeric differences in the approximation with model PSFs.
Screening of the PPS processed data has taken place with fully satisfactory results. The test pipeline version is expected to evolve into the final PPS based on SAS 16, which should become operational in the first weeks of 2017.

3.3.5 XCal data

The new SAS version SASv16.0 was compared with its previous version SASv15.0 doing spectral extractions of 288 individual exposures of 51 targets of various source types throughout all epochs of the mission which are included the XMM-Newton SOC cross-calibration archive:

- 178 exposures of on-axis point targets of various source types, mainly with continuum dominated spectra, from isolated neutron stars to AGN.
- 74 exposures of 2 different thermal supernova remnants
- 12 exposures 4 different galaxy clusters
- 24 off-axis point sources (AGN).

All comparison results of the spectral extractions using the two different versions of the SAS show conformity according to the calibration status and its software support of the corresponding version.

3.4 Processing with SAS built on a different flavour than the one used - Compatibility

The two Linux versions (RedHat EL 6.8 and Ubuntu 16.04) have been used by following Linux OSs for checking the compatibility when running SAS binaries built on a different Linux version:

- SAS 16 xmmsas_20161216_1833 built on RHEL6.8 64-bit installed in:
  - Ubuntu 16.04LTS (native perl 5.22.1): All Ok.
  - Fedora 24 64-bit (native perl 5.22.2). All OK.
  - OpenSUSE Leap 42.1 64-bit (native perl 5.18.2). Problems with perl. We had to install 5.22.2.
  - CentOS 6.8 64-bit (native perl 5.10.1). Problems with perl. We had to install 5.22.2.
  - CentOS 7.2 64-bit (native perl 5.16.3). Problems with perl. We had to install 5.22.2.
- SAS 16 xmmsas_20161216_1833 built on Ubuntu 16.04LTS 64-bit installed in:
  - Ubuntu 16.04LTS (native perl 5.22.1): All Ok.

3.5 Dedicated analysis

3.5.1 EPIC data - G21.5-0.9

The non-thermal SNR G21.5-0.9 has been used as one of the standard targets for the validation of the EPIC effective area calibration. Additionally, this source has proven useful in multi-mission cross-calibration studies (Tsujimoto et al. 2011). Its spectrum can be well modelled by a simple power-law combined with a photoelectric absorption.
In observation 0122700101, G21.5-0.9 was observed with MOS in Full Frame mode and PN in Extended Full Frame Mode (all using Medium Filter) for 30 ks. MOS and PN source spectra were extracted from a circular region ($\sim$ 2.5 arcmin radius) around the SNR, and spatially filtered through their common exposure mask. MOS background spectra were obtained from annular regions around the source, whereas PN background was obtained from neighbouring source free regions. EPIC data were reduced with SAS 16.0 and spectra were extracted with standard event pattern selection.
The results of the comparison of PN and MOS are summarised in Fig. 1, and are essentially in agreement with the previous SAS science validation study. This is as expected, as no significant changes in energy scale calibration have been introduced in the meantime.

3.5.2 EPIC Data - PKS 0558-504

PKS 0558-504 is a well studied radio loud Narrow Line Seyfert 1 galaxy (e.g. Siebert et al. 1999), and has been observed by *XMM-Newton* as calibration target. Its 2-10 keV spectrum is characterised by a spectral slope $\Gamma \sim 2.2$, and the 0.2-2 keV emission is dominated by a large and featureless soft excess.
EPIC spectra of ObsIds 0125110101 and 0129360201 are compared as part of this science validation. In the exposures compared here, PN was operated in Small Window Mode, and both MOS instruments in Large Window Mode. Data were reduced with SAS 16.0 (using the respective latest calibration files), and resulting spectra were fit in the 0.3–10 keV band with a

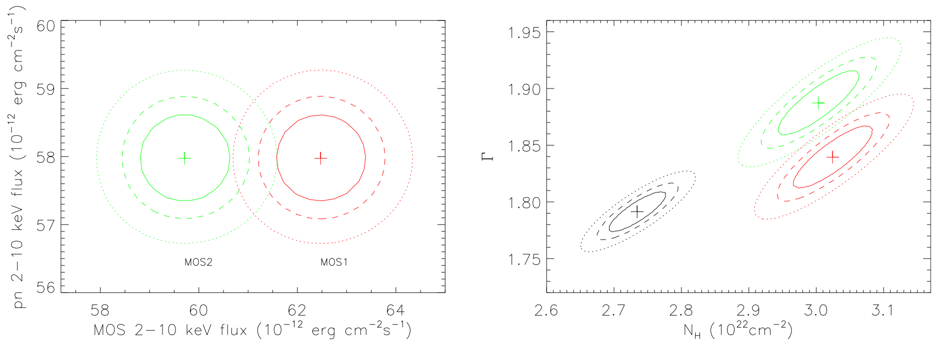

Figure 1: Comparison of PN versus MOS spectral fits of G21.5-0.9. Spectra based on data reduced with SAS 16.0, using standard pattern selection. *Left panel*: the 2-20 keV flux confidence contours for PN versus MOS1 (red) and MOS2 (green). *Right panel*: column density versus photon index confidence contours for PN (black), MOS1 (red), MOS2 (green). Levels shown are at 68%, 90% and 99% confidence.

model consisting of a power-law and bremsstrahlung component with an ISM absorbtion model (Papadakis et al. 2010). As the MOS data are subject to pile-up the spectra were extracted from annular regions with core exclusion radii of 10".
The best fit results are summarised in Tables 3 and 4. The main differences between instruments are due to the imperfect relative effective area calibration, resulting in fluxes which are formally not consistent across all three instruments.

Table 3: Comparison of MOS and PN spectral fits to PKS 0558-504 (ObsId 0125110101) with a power-law plus bremsstrahlung model.

Instrument	kT keV	Γ	Flux 10^{-11} erg cm^{-2} s^{-1} (0.3–2.0 keV)	Flux 10^{-11} erg cm^{-2} s^{-1} (2.0–10.0 keV)	χ^2/d.o.f.
MOS1	$0.24^{+0.06}_{-0.05}$	$2.15^{+0.13}_{-0.15}$	$2.08^{+0.02}_{-0.02}$	$0.95^{+0.05}_{-0.06}$	251.8/273
MOS2	$0.31^{+0.07}_{-0.08}$	$2.12^{+0.16}_{-0.19}$	$2.26^{+0.02}_{-0.02}$	$0.92^{+0.05}_{-0.05}$	250.1/258
PN	$0.28^{+0.03}_{-0.03}$	$2.21^{+0.06}_{-0.06}$	$2.46^{+0.03}_{-0.03}$	$1.06^{+0.07}_{-0.06}$	529.8/528

3.5.3 Standard tests of `esas`

Due to the special characteristics of `esas`, the package for analysis of extended sources observed with the EPIC cameras, particular validation tests are run with every new SAS version, to ensure its integrity and the validity of their separate calibration files. With this purpose the full thread for `esas` images extraction (http://www.cosmos.esa.int/web/xmm-newton/sas-thread-esasimage) has been run on the odf 0097820101 (Abel 1795 cluster), with both PN and MOS data.
A first run performed in the first iteration test showed that the task `cheese` was missing. An SPR has been raised and the problem rapidly solved. Further tests have shown no discrepancies to earlier results.
In SASv16 the parameter *chisqfl* has been removed from `mos_back`, `pn_back`, but these are not used in the calls used during the testing procedure. Nevertheless, checks have shown that the

Table 4: Comparison of MOS and PN spectral fits to PKS 0558-504 (ObsId 0129360201) with a power-law plus bremsstrahlung model.

Instrument	kT	Γ	Flux		χ^2/d.o.f.
	keV		10^{-11} erg cm^{-2} s^{-1}		
			(0.3–2.0 keV)	(2.0–10.0 keV)	
MOS1	$0.39^{+0.09}_{-0.08}$	$1.98^{+0.14}_{-0.26}$	$2.49^{+0.02}_{-0.02}$	$1.32^{+0.06}_{-0.06}$	317.7/338
MOS2	$0.25^{+0.10}_{-0.05}$	$2.25^{+0.10}_{-0.10}$	$2.46^{+0.02}_{-0.02}$	$1.17^{+0.05}_{-0.05}$	327.7/337
PN	$0.36^{+0.03}_{-0.03}$	$2.10^{+0.04}_{-0.05}$	$2.42^{+0.01}_{-0.01}$	$1.17^{+0.02}_{-0.02}$	715.5/705

removal of the parameter has no impact on running the tasks. The parameter does not appear when the tasks are called with the -p or -d options.

3.5.4 General comments on SAS OM data processing

There are no major changes in the OM tasks in this version of SAS, except those related to adaptation to the new Fortran GNU compiler. They are related to initialisation of arrays and variables.

The validation of the OM tasks built with the new compiler has uncover a bug in the task `omdetect`. Correcting for this bug has produced some differences between, let's say SAS 15 and SAS 16, which are not due to the change of compilers (see below). In fact, the change of compilers presents no differences among the two systems: most of the results obtained after processing OM data with SAS are almost identical.

In addition to checking the mentioned new implementations, as in previous deliveries, this Science validation has been devoted to check and confirm that the main functionalities already present in previous versions are maintained, in other words, we confirm the overall stability of the system. The reader is thus referred to previous Science Validation Reports for more detailed comments or descriptions.

3.5.5 `omdetect`

The task `omdetect` is used in the processing of all types of OM data: image photometry, fast mode time series and grism data extraction. The task looks for sources, measures their position (in detector coordinates) and performs aperture photometry. In the case of grism data, zero and first orders are identified for ulterior spectral extraction.

For image data `omdetect` is run twice in the `omichain`, first on the observed images from which most of the sources are measured, and then when a given filter is used in several exposures, the corresponding images are mosaiced and co-added and a second run of detection is done with the aim of finding fainter sources not detected in the single exposure image. Prior to being co-added, the images are north aligned (rotated), so that the second detection is done on "sky" images, namely rotated and then added together.

The corrected `omdetect` task finds more sources on the second detection than the previous versions of the task. These additional sources are not only due to the increase in signal to noise one could expect from co-adding the images, but mainly to a side effect of the rotation. Rotating the images implies some kind of interpolation and resampling of the original data (pixels) so that some very low level features become enhanced and may be identified by the task as real sources.

After running tests on 50 different observations we see that the new task finds around 7 % more sources than the previous version. We have cross-identified the new additional sources (nearly 2000) with several catalogs as GAIA DR1 and USNO-B1.0. Using a tolerance of 2 arcsec only 30 % give a successful match. If the tolerance is relaxed to 5 arcsec then we obtain 50 % of matches.
When the second detection was implemented in `omichain` the additional sources were assigned a flag indicating that their photometric quality was inferior to the one obtained in normal single images, because coincidence loss cannot be properly corrected in the former. Now this flag shall also indicate that the whole reliability of these sources is dubious.
We should note that fast mode data are not affected by this problem since the detection is done in non rotated images.
In grism data, the images are rotated to align the dispersion direction with the Y-axis and then the search of the spectra is performed. Since the characterisation of the spectra is based in the cross-identification of zero and first orders, the effect of the rotation in the detection is somehow compensated and in any case the old and the new systems both perform the detection on rotated images. There is no effect on the final extracted spectral data which are almost identical.

3.5.6 Repeatability of OM filter photometry

Several spectrophotometric standard stars are observed repeatedly with OM in order to establish and monitor the photometric and flux calibrations. These are the white dwarfs GD 153, HZ 2 and BPM 16274.
As we have done in the past all existing data of these stars have been reprocessed using SAS 16 (with the new Fortran compiler).
The results are presented in Table 5. For comparison we give also these results obtained with SAS 15 (see Table 6.
The quoted errors are the standard deviation of the mean values given as percentage.
We see that after all corrections are applied, the count rates of these stars obtained from all observations taken during the life of OM do not vary within 3 %.

Table 5: Standard stars processed with SAS 16.0 (GNU Fortran): average count rates of several observations

star	N_obs	UVW2	UVM2	UVW1	U	B	V
GD153	14	83.12	161.78	329.49	420.18	283.45	71.35
error (%)		1.4	1.5	0.8	1.4	1.0	2.1
HZ2	17	23.76	48.27	111.73	168.84	148.83	43.73
error (%)		2.0	1.3	1.3	0.9	0.8	2.9
BPM16274	32	14.73	30.34	72.92	112.68	107.77	32.95
error (%)		1.7	1.2	1.0	0.8	0.8	2.2

3.5.7 Caveats of OM related tasks in previous versions of SAS

Some little bugs and caveats that were reported in previous versions of SAS remain unsolved (e.g. in Fast Mode processing or in the interactive tasks `omsource` and `omgsource` in SAS 15). The reader is referred to previous Science Validation reports.

Table 6: Standard stars processed with SAS 15.0: average count rates of several observations

star	N_obs	UVW2	UVM2	UVW1	U	B	V
GD153	13	83.51	163.59	331.00	420.92	284.60	71.53
error (%)		1.6	1.6	1.0	1.4	1.0	2.4
HZ2	17	23.90	48.78	112.30	169.29	149.36	43.88
error (%)		2.2	2.2	1.5	0.9	0.9	3.2
BPM16274	31	14.80	30.65	73.25	113.02	108.11	33.05
error (%)		1.8	1.9	1.1	0.7	0.9	2.5

4 New and updated in SAS16 - Validation

4.1 New in SAS 16 - `evqpb`

Analysis of odf 0555630101 has been performed for testing the new task `evqpb` under SAS v16.0. The output products (for all three EPIC cameras) have been inspected and the log files checked for errors.
In the first iteration, it was realised that two columns, present in the CCF (REV and WEIGHT) were not propagated in the output, what would be convenient. Following a corresponding report asking for this it has been implemented. Also the exposure weight per event according to CCD had to be included. This is necessary to produce already scaled images and spectra with evselect. Not providing exposure scaled images and spectra would make it very complicated for the user to do it according to CCD. The user would have to use non standard tasks to do it, or even write their own code. The inclusion of EWEIGHT avoids this.
The test performed had three steps:

- Test the validity of the new CCFs with the FWC information. No problems found. The CCF contains in a second extension information if a particular MOS CCD is noisy. The SAS task evqpb at present has hardcoded not to use MOS noisy CCDs as flagged in the CCF. In a future release we should let the user decide whether to include noisy CCDs or not via task parameter.

- Test on the performance of the task.

  – Using large exposure factors up to the maximum allowed exposure factor. Use EPIPC pn for the test.
  `evqpb` *table*=1594_0555630101_EPN_S003_ImagingEvts.ds *exposurefactor*=1000.0 *attfile*=1594_0555630101_AttHk.ds *outset*=evqpb_pnevqpb_pn_fullinterval1.fits
  Results:
  evqpb:- CCD 1 EXPOSURE 34360.6 TARGET 3.43606e+07 TOTAL EXP 453123 Weight Factor 0.0758307
  evqpb:- CCD 2 EXPOSURE 34286.6 TARGET 3.42866e+07 TOTAL EXP 453057 Weight Factor 0.0756783
  evqpb:- CCD 3 EXPOSURE 34221 TARGET 3.4221e+07 TOTAL EXP 453002 Weight Factor 0.0755426
  evqpb:- CCD 4 EXPOSURE 34436.5 TARGET 3.44365e+07 TOTAL EXP

454185 Weight Factor 0.0758204
evqpb:- CCD 5 EXPOSURE 34370.2 TARGET 3.43702e+07 TOTAL EXP 454086 Weight Factor 0.0756909
evqpb:- CCD 6 EXPOSURE 34291.2 TARGET 3.42912e+07 TOTAL EXP 454008 Weight Factor 0.0755298
evqpb:- CCD 7 EXPOSURE 34232.5 TARGET 3.42325e+07 TOTAL EXP 451458 Weight Factor 0.0758265
evqpb:- CCD 8 EXPOSURE 34161.1 TARGET 3.41611e+07 TOTAL EXP 451371 Weight Factor 0.0756829
evqpb:- CCD 9 EXPOSURE 34098.3 TARGET 3.40983e+07 TOTAL EXP 451311 Weight Factor 0.075554
evqpb:- CCD 10 EXPOSURE 34346.3 TARGET 3.43463e+07 TOTAL EXP 452812 Weight Factor 0.0758511
evqpb:- CCD 11 EXPOSURE 34268.5 TARGET 3.42685e+07 TOTAL EXP 452717 Weight Factor 0.0756953
evqpb:- CCD 12 EXPOSURE 34197.7 TARGET 3.41977e+07 TOTAL EXP 452664 Weight Factor 0.0755477

`evqpb` *table*=1594_0555630101_EPN_S003_ImagingEvts.ds *exposurefactor*=10000.0 *attfile*=1594_0555630101_AttHk.ds *outset*=evqpb_pn_fullinterval2.fits
Results:
evqpb:- CCD 1 EXPOSURE 34360.6 TARGET 3.43606e+08 TOTAL EXP 453123 Weight Factor 0.0758307
evqpb:- CCD 2 EXPOSURE 34286.6 TARGET 3.42866e+08 TOTAL EXP 453057 Weight Factor 0.0756783
evqpb:- CCD 3 EXPOSURE 34221 TARGET 3.4221e+08 TOTAL EXP 453002 Weight Factor 0.0755426
evqpb:- CCD 4 EXPOSURE 34436.5 TARGET 3.44365e+08 TOTAL EXP 454185 Weight Factor 0.0758204
evqpb:- CCD 5 EXPOSURE 34370.2 TARGET 3.43702e+08 TOTAL EXP 454086 Weight Factor 0.0756909
evqpb:- CCD 6 EXPOSURE 34291.2 TARGET 3.42912e+08 TOTAL EXP 454008 Weight Factor 0.0755298
evqpb:- CCD 7 EXPOSURE 34232.5 TARGET 3.42325e+08 TOTAL EXP 451458 Weight Factor 0.0758265
evqpb:- CCD 8 EXPOSURE 34161.1 TARGET 3.41611e+08 TOTAL EXP 451371 Weight Factor 0.0756829
evqpb:- CCD 9 EXPOSURE 34098.3 TARGET 3.40983e+08 TOTAL EXP 451311 Weight Factor 0.075554
evqpb:- CCD 10 EXPOSURE 34346.3 TARGET 3.43463e+08 TOTAL EXP 452812 Weight Factor 0.0758511
evqpb:- CCD 11 EXPOSURE 34268.5 TARGET 3.42685e+08 TOTAL EXP 452717 Weight Factor 0.0756953
evqpb:- CCD 12 EXPOSURE 34197.7 TARGET 3.41977e+08 TOTAL EXP 452664 Weight Factor 0.0755477

If exposure factors larger than the available FWC exposure are used, the total FWC exposure available is used. So using exposurefactor=1000.0 and exposurefactor=10000.0 has the same effect.

– Using a time interval smaller that the science exposure

`evqpb` *table*=1594_0555630101_EPN_S003_ImagingEvts.ds *exposurefactor*=0.5 *attfile*=1594_0555630101_AttHk.ds *outset*=evqpb_pn_smallinterval.fits

evqpb:- CCD 1 EXPOSURE 34360.6 TARGET 17180.3 TOTAL EXP 17233.3 Weight Factor 1.99384
evqpb:- CCD 2 EXPOSURE 34286.6 TARGET 17143.3 TOTAL EXP 17228.5 Weight Factor 1.99011
evqpb:- CCD 3 EXPOSURE 34221 TARGET 17110.5 TOTAL EXP 17224.6 Weight Factor 1.98675
evqpb:- CCD 4 EXPOSURE 34436.5 TARGET 17218.3 TOTAL EXP 17305.2 Weight Factor 1.98996
evqpb:- CCD 5 EXPOSURE 34370.2 TARGET 17185.1 TOTAL EXP 17300.7 Weight Factor 1.98664
evqpb:- CCD 6 EXPOSURE 34291.2 TARGET 17145.6 TOTAL EXP 17294.2 Weight Factor 1.98281
evqpb:- CCD 7 EXPOSURE 34232.5 TARGET 17116.2 TOTAL EXP 17165.3 Weight Factor 1.99428
evqpb:- CCD 8 EXPOSURE 34161.1 TARGET 17080.5 TOTAL EXP 17159.2 Weight Factor 1.99083
evqpb:- CCD 9 EXPOSURE 34098.3 TARGET 17049.2 TOTAL EXP 17155.4 Weight Factor 1.98762
evqpb:- CCD 10 EXPOSURE 34346.3 TARGET 17173.1 TOTAL EXP 17240.7 Weight Factor 1.99216
evqpb:- CCD 11 EXPOSURE 34268.5 TARGET 17134.3 TOTAL EXP 17234.8 Weight Factor 1.98833
evqpb:- CCD 12 EXPOSURE 34197.7 TARGET 17098.9 TOTAL EXP 17231 Weight Factor 1.98467

If smaller factors are used the result is as expected and the weights per CCD are a factor of 2. That is, you have to weight the FWC exposure by a factor 2 to make up for the science exposure.

4.2 Upgraded in SAS 16.0: `ebkgreg`

The task generates background regions which are dependent on the camera, the observing mode and the position of the source in the field of view. Somewhat counter-intuitively it does not look for a source-free region of sky. A user, or the pipeline, then has to create the background from an event file where sources have been previously removed. This is now explained in the user documentation.
Testing: the task was run on the observation 0147511601, where the EPIC-pn and MOS-1 cameras were run in Full Frame mode.
Three steps constituted the testing:

- Test 1: Individual source at RA: 163.412 Dec: 57.516 on the EPIC-pn ebkgreg *imageset*=P0147511601PNS003IMAGE_8000.FITS *withsrclist*=no *withcoords*=yes *x*=163.412 *y*=57.516 *coordtype*=EQPOS *r*=60

Returns a background region:

```
Best background extraction region estimate:
  Extraction region shape    : CIRCLE
  RA, Dec (deg.)             : 163.36, 57.5637
  Extraction radius (arcsec) : 101.439
```

which fits neatly into the same CCD as the source and correctly avoids contamination by the source itself.

- Test 2: Individual source near the edge of the MOS-1 camera at RA: 163.525 Dec: 57.588

  ebkgreg *imageset*=P0147511601M1S001IMAGE_8000.FITS *withsrclist*=no *withcoords*=yes *x*=163.525 *y*=57.588 *coordtype*=EQPOS *r*=60

  This returns a background region:

```
Best background extraction region estimate:
  Extraction region shape    : CIRCLE
  RA, Dec (deg.)             : 163.653, 57.5597
  Extraction radius (arcsec) : 168
```

  which lies within the same CCD as the source but mainly outside the field-of-view.

- Test 3: An EPIC-pn source list

  This runs on the final source list, combined across the instruments, that the pipeline produces. A user can generate this source list using `edetect_chain` and then the SRCMATCH command. Here we use the archive version:
  ebkgreg *srclisttab*=P0147511601EPX000OBSMLI0000.FIT
  imageset=P0147511601PNS003IMAGE_8000.FITS

  Background regions are written into the source list and look sensible.

4.3 New in SAS 16.0: `edetect_stack`, a new meta-task for source detection by overlapping observations

This meta-task is still at experimental level and, while extensive validation will be only in the future possible, in the framework of production of a new catalogue, testing with a limited number of datasets was part of this validation.
For this purpose, two different tests were designed and applied:

- processing with `edetect_stack` an observation performed in mosaic mode with large overlap: to this purpose, the observation of Jupiter OBSID 0200080701 has been taken. The overlap between the four positions is at the level of 70%. We have ran in parallel source detection also with `emosaic_proc`. Both metatasks use the same basis tasks for source detection (`eboxdetect` and `emldetect`, although with different parameters), but different ways of calculating the background. The coincidence of detected sources by a total of ≈ 110 sources in the field was at the 90% level. `edetect_stack` showed reliability in this case.

- processing 11 individual observations of the same field: the large number of observations stacked caused problems to `edetect_stack`. The call to `emldetect` within the metatask could not get to the end, even after 2 weeks of running with 100% of a CPU, and had to be stopped. Since the same input as provided within the metatask was used for running `emldetect` with its standard parameters, and the processing finished correctly after 8 hours, the problem with `edetect_stack` must surely be with the (many) non-default parameters used in its call.

Although the test was partly negative, it was outside the scope of this validation to do a thorough analysis of a meta-task (and the parameters used for the calls to the diverse tasks embedded), which is considered still experimental at this stage.

4.4 New in SASv16.0: Correction to the RGS Effective Area

The indication of a systematic departure between fluxes derived from RGS1 and RGS2 data has led to the computation of an empirical correction to the RGS effective area, following the work by Kaastra, de Vries and den Herder (2015). This correction has been incorporated as new extensions in the EFFAREACORR CCF, for RGS1 and RGS2, first and second order data. A description of the methodology and the implementation is given in the work mentioned above and in the Release Note for the updated CCF (XMM-CCF-REL-340).
The application of this correction results in an accuracy of the first order effective area of 1-2% for RGS1 and RGS2, respectively, over most of the wavelength range. Also, there is a substantial improvement in the calibration of the second order, see Fig. 4.4.

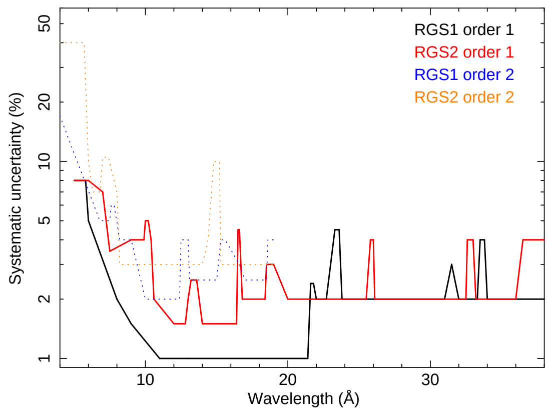

Figure 2: Estimated calibration uncertainty of the effective area after the application of this correction (from Kaastra et al. 2015).

Table 7 shows the best fit parameters and their errors for a joint fit of RGS1 and RGS2 first order spectra of the observations of PKS 0558-504 in rev. 1603 and of Mkn 421 in rev. 3096, with and without this correction. While the values of the parameters are similar, there is a

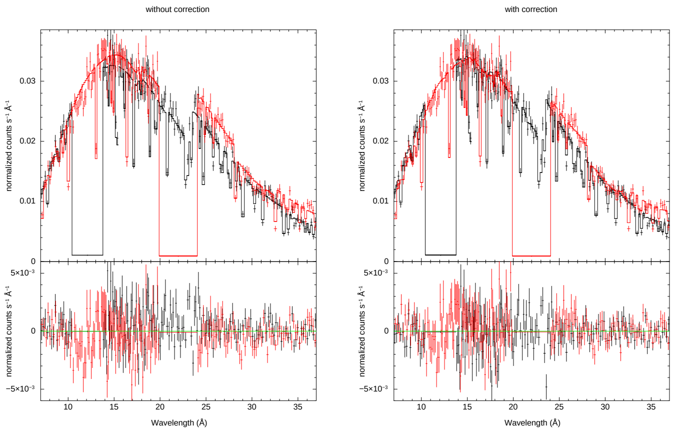

Figure 3: Test case: fit to the observation of PKS 0558-504 in rev. 1603

statistical improvement in the fit when applying this correction, with C decreasing from 5214 to 5183 for PKS 0558-504, and from 5241 to 5072 for Mkn 421.

Table 7: Best fit parameters for a joint fit of RGS1 and RGS2 first order spectra

	PKS 0558-504 in rev. 1603	
	without correction	with correction
N_H (10^{20})	3.11 ± 0.29	3.22 ± 0.29
slope	2.66 ± 0.03	2.66 ± 0.03
normalisation (10^{-3})	6.93 ± 0.07	7.18 ± 0.07
C/dof	5214/4981	5183/4981
	Mkn 421 in rev. 3096	
	without correction	with correction
N_H (10^{20})	2.15 ± 0.30	2.38 ± 0.30
slope	2.62 ± 0.03	2.62 ± 0.03
normalisation (10^{-2})	4.00 ± 0.04	4.11 ± 0.04
C/dof	5241/5069	5072/5069

There is also a significant improvement when comparing independent fits to RGS1 and RGS2 data. After applying the effective area correction, the discrepancies between both instruments decrease substantially, as shown in Table 8.
The application of this correction is implemented in SASv16 as a non-default option to `rgsproc` and `rgsrmfgen`, through the switch
`witheffectiveareacorrection=yes`.

4.5 RGS2 observations in Single-Node-Readout mode

The filtering of events "ON_NODE_INTERFACE" has been suppressed for RGS2 observations taken in "Single-Node-Readout" mode (this mode was implemented in revolution 1408). The figure shows the difference between filtering out or not events having the flag "ON_NODE_INTERFACE".

Table 8: Comparison of parameters for independent fits to RGS1 and RGS2 first order spectra

	PKS 0558-504 in rev. 1603		Mkn 421 in rev. 3096	
	without correction	with correction	without correction	with correction
ratio of N_H	1.14	0.91	1.25	0.84
difference in slope	-0.04	0.02	0.07	-0.01
ratio of normalisations	1.04	0.97	1.03	0.97

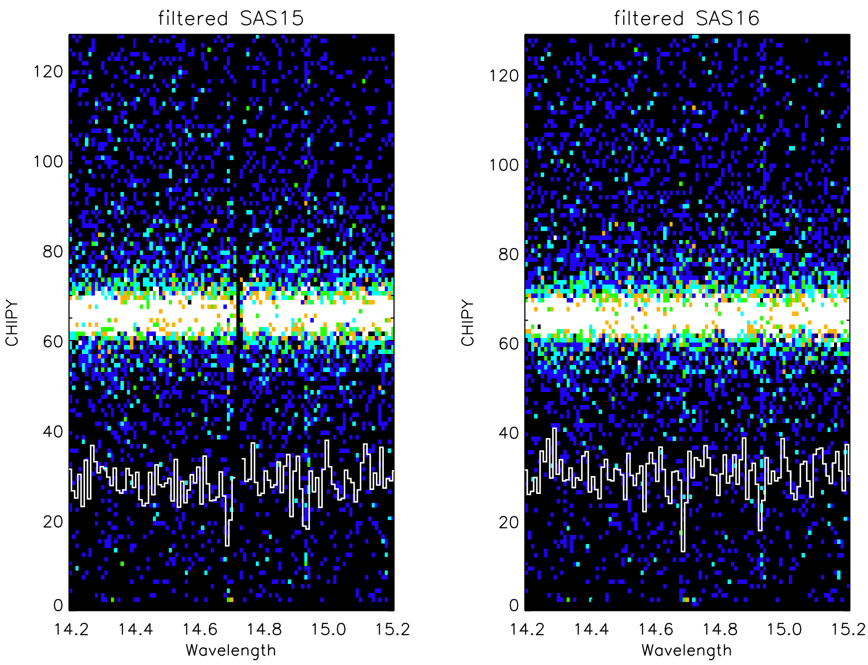

Figure 4: Differences in the filtering of a RGS2 observation taken in "Single-Node-Readout" mode (left, with SASv15; right, with SASv16). Shown is CCD6 in the filtered event list of obsid 0414190401 (3C 273 in rev 1465). The node interface is near 14.7Å. Overplotted on the images are the corresponding RGS2 first order fluxed spectra.

4.6 Processing of observations taken in "Multi-Pointing Mode"

RGS observations taken in "Multi-Pointing-Mode" were wrongly processed in SASv15. A too strict comparison of the values of the keyword "DATAMAX" between the two nodes of the chips made in `rgspectrum` was marking an excessive number of channels as "dubious", and hence flagging them with `QUALITY=1` in the extracted count spectrum.
With the suppression of this check, all channels close to a node interface are marked as valid, though with a reduced exposure, as expected (see Fig. 5).

4.7 New in SAS 16 - use of `embadpixfind` for RGS bad pixel / column finding

The method used in SAS to detect hot RGS columns (`rgsbadpix`) is based on an algorithm in which each CCD column is compared to the adjacent ones (RGS-COL-CAL-00015). This algorithm is controled by two parameters, `colsharpness` and `colnoiselimit`, which have default values of 8 and 250, respectively.
It has been found that bright, sharp emission lines are occasionally flagged as "hot columns" by

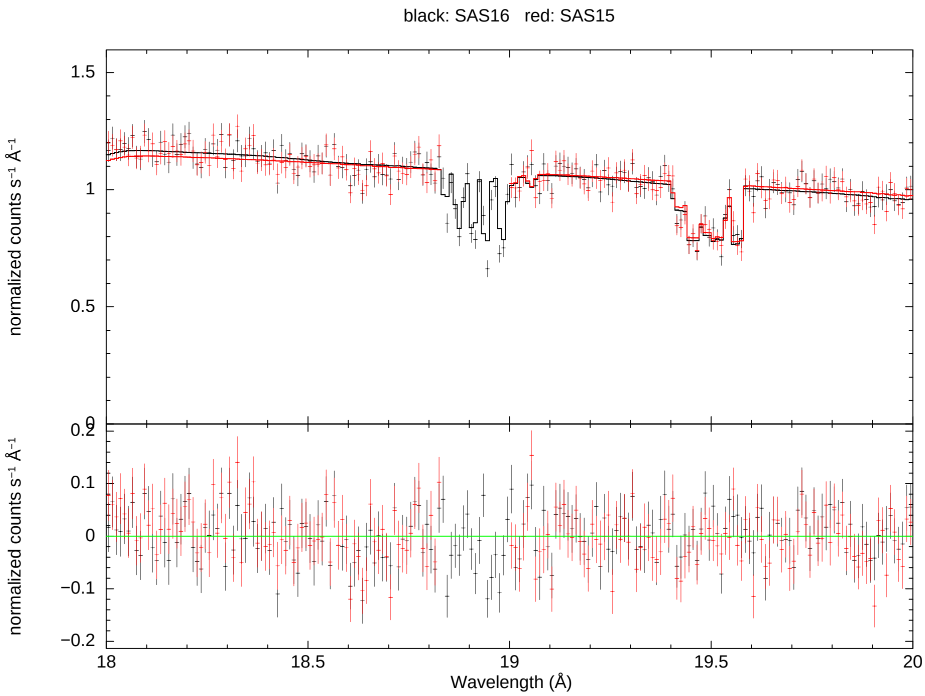

Figure 5: Example of the processing with SAS15 and SAS16 of an RGS1 observation of Mkn 421 taken in multipointing mode. This observation consisted on five separated pointings at different offsets along the dispersion direction. The processing with SAS15 (in red) resulted in a large gap around the node interface near 19Å. When processing with SAS16 (in black) the gap disappears, as all the channels are marked as valid. As expected, due to the multi-pointing strategy, the exposure is lower in these channels, but the effective is computed correctly (see the residuals to the fit to a power-law in the bottom panel). The processing is also correct for the channels close to a bad column at 19.5Å.

this algorithm.

We have checked how often this problem happens examining the spectra of the bright wavelength calibrators Capella, AB Dor, HR 1099 and Procyon, 88 observations in total.

The `BADPIX` extensions in the filtered event files have been examined for hot columns around the predicted position of bright emission lines (±0.05 Å), see Table 9, left - black - column.

Most of the lines incorrectly flagged as hot columns correspond to the brightest of the four targets, Capella: the Fe XVII 17.05 Å line is flagged only in Capella, and the O VIII 18.97 Å line in Capella and in eight cases (four in RGS1, four in RGS2) in AB Dor and HR 1099. The two brightest lines in Capella are flagged in ≈ 40% of the RGS1 spectra. The fraction is lower for RGS2, 25%.

AB Dor and HR 1099 show similar statistics: only four and three lines are flagged in RGS1 spectra, respectively. In RGS2 five lines are flagged in the spectra of AB Dor, and seven in those of HR 1099. In the faintest of the four targets, Procyon, only the C VI line at 33.7 Å is flagged, once in RGS1.

We can then conclude that the problem is very minor if we take into account the bulk of XMM observations, but that it may have an impact in the case of **very** bright objects with emission-line spectra.

A test was made to check whether this problem might be solved using the same bad columns finding algorithm, but with the value of the parameter `colsharpness` set to 9 Table 9, central

Table 9: Number of lines flagged as "hot" by the three methods discussed in the text `rgsbadpix` / `rgsbadpix colsharpness=9` / `embadpix`

	Capella			AB Dor			HR 1099			Procyon			Total		
n. obs	28			29			26			5			88		
RGS1															
15.02Å	9	4	0	0	0	0	0	0	0	0	0	0	9	4	0
16.78Å	3	0	0	0	0	0	1	1	1s	0	0	0	4	1	1
17.05Å	12	6	0	0	0	0	0	0	1s	0	0	0	12	6	1
18.97Å	10	5	1s	3	0	0	1	0	0	0	0	0	14	5	1
21.60Å	0	0	1s	0	0	1s	1	1	2s	0	0	0	1	1	4
21.80Å	1	0	7s	0	0	6s	0	0	0	0	0	0	1	0	13
33.73Å	2	1	1s	1	1	0	0	0	0	1	0	0	4	2	1
all	38	16	10	4	1	7	3	2	4	1	0	0	45	19	21
RGS2															
15.02Å	6	6	1	0	0	0	0	0	0	0	0	1	6	6	2
16.78Å	3	1	0	1	1	0	0	0	0	0	0	0	4	2	0
17.05Å	6	5	0	0	0	0	0	0	0	0	0	0	6	5	0
18.97Å	7	2	0	4	4	1s	0	0	0	0	0	0	11	6	1
33.73Å	1	0	1s	0	0	1	7	7	7s	0	0	0	8	7	9
all	23	14	2	5	5	2	7	7	7	0	0	1	35	26	12

s: segments

- blue - column.
Taking Capella as a "worst case" scenario, the use of `colsharpness=9` reduces the number of flagged lines to nearly half, from 38 to 16 in RGS 1, from 23 to 14 in RGS2.
In the case of RGS1 the number of lines flagged is reduced by 50%. In RGS2 the difference is smaller. Limited tests have shown that larger values of `colsharpness` might improve this result.
A new algorithm, adapted from the one used in the MOS cameras for hot pixels search, has been implemented in SAS (`embadpixfind`). The results of processing this same sample with this method are shown in the right - red - column in Table 9.
With this method the number of lines flagged is substantially reduced. `'embadpixfind'` suppresses the flagging of most of the brightest lines, but flags the RGS1 faint 21.6 and 21.8 Å lines in several spectra (eight cases in Capella, seven in AB Dor, two in HR 1099). With this method, in most cases segments rather than full columns are flagged (see below).

4.7.1 General comparison of both methods

To asess the applicability of this new method to other type of spectra, a sample of 177 observations of sources with continuum spectrum has been processed with two options: the default SAS setting (`rgsbadpix colsharpness=8`) and `embadpixfind`. Since the parameters of the latter method are not configurable in the current implementation, the processing whose results are discussed here has been made with default values of widths and thresholds, that are the same used for the MOS cameras.
`embadpixfind` flags twice as many bad columns and pixels as `badpixfind` (see Table 10). The number of full columns flagged is similar, hence the difference is in the number segments.
Figure 6 shows the distribution of the lengths of the flagged segments. While `rgsbadpix` finds very few segments with lengths between 10 and 90 pixels, segments found with `embadpix` cover

Figure 6: Length of flagged segments

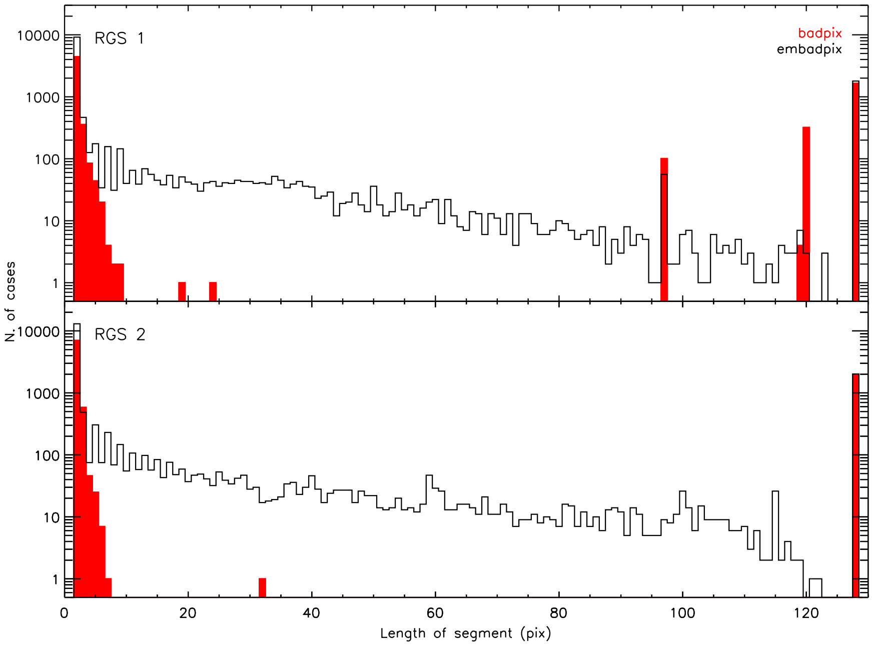

Table 10: Comparison of `rgsbadpix` and `embadpixfind`

	number of hot segments and columns		number of hot columns	
	rgsbadpix	embadpixfind	rgsbadpix	embadpixfind
RGS1	7056	14245	1971	1802
RGS2	9726	18883	2008	2021

continuously all the range of lengths between 2 and 128 pixels.

4.7.2 Usage of "hot columns" algorithms

The use of `embadpixfind` for the detection of "hot columns" in the case of spectra with very bright emission-line represents an improvement with respect to the default method, `rgsbadpix`, while there are no differences for fainter spectra.
The differences in the algorithms makes `embadpixfind` to detect segments of all lengths, while `rgsbadpix` detects only either full columns or segments shorter than ≈ 10 pixels.
To evaluate the effect of this different behaviour and assess the applicability of the method to spectra of different types and intensities, a more complete exploration of the parameter space is necessary. Therefore, we recommend this method only in case the user suspects that the profile of a bright emission line is distorted due to the detection of a close-by hot column.

5 Conclusion

The SAS scientific validation process concluded that SAS 16 was validated and should be released, as it happened on January 16, 2017.

References

Kaastra , J., de Vries, C., den Herder, J.W., SRON-RGS Internal Report, August 2015
Ojero, E., "XMM-Newton SAS Integration", v1.1, XMM-SOC-SW-TN-0034, September 2016
Papadakis, I.E. et al. 2010, A&A 510, A65
Sanz-Forcada, J., Micela, G., & Maggio, A. 2007, in XMM-Newton: The Next Decade, p3
Siebert, J. et al. 1999, A&A 348, 678
Tsujimoto, M. et al. 2011, A&A 525, A25