
Plan d'essais MAJA

Release 4.3.0

MAJA Team

Dec 16, 2020

GLOSSAIRE ET LISTE DES PARAMÈTRES AC & AD 1

1	Glossaire et Liste des paramètres AC & AD	5
2	Généralités	7
2.1	Documents de référence	7
2.2	Documents applicables	7
3	introduction	9
3.1	objet du document	9
3.2	structure du document	9
4	Environnement de validation	11
4.1	Configuration matérielle et dimensionnement	11
4.2	Moyens de validation	11
4.3	Outils	11
4.3.1	Moniteur de tests	11
4.3.2	Exploitation des résultats des tests : Serveur CDash et SonarCube	12
4.4	Données de Validation	12
4.4.1	Venus	12
4.4.2	Cas Venus (au format “muscate”)	12
4.4.3	Sentinel 2 (S2A et S2B)	12
4.4.4	Cas Sentinel2 (au format “muscate”)	13
4.4.5	Cas Landsat 8 (au format MUSCATE)	13
4.4.6	Cas Landsat 8 (au format “natif”)	13
4.5	Installation de l’environnement de validation	13
4.5.1	Pré-requis	13
4.5.2	Lancement des tests	13
5	LES TESTS UNITAIRES	15
5.1	Objectifs	15
5.2	Stratégie et méthode des tests unitaires	15
5.3	Formalisation des tests unitaires	16
6	LES TESTS DE VALIDATION ALGORITHMIQUE	17
6.1	Objectifs	17
6.2	Stratégie et méthode des tests numériques	17
6.3	Formalisation des tests de validation algorithmique	18
6.4	LISTE DES TESTS DE VALIDATION ALGORITHMIQUE	18
6.5	DESCRIPTION DÉTAILLÉE DES TESTS DE VALIDATION ALGORITHMIQUE	19
6.5.1	apTvAngleListToImage	19
6.5.1.1	Objectif	19
6.5.1.2	Description	20

6.5.1.3	Liste des données d'entrées	20
6.5.1.4	Liste des produits de sortie	20
6.5.1.5	Prérequis	20
6.5.1.6	Durée attendue	20
6.5.1.7	Epsilon utilisé sur la non regression	20
6.5.1.8	Vérifications à effectuer	20
6.5.1.9	Mise en oeuvre du test	20
6.5.1.10	Journal d'essai de la recette	20
6.5.1.10.1	Notes sur le déroulement du test	20
6.5.1.10.2	Conclusion du déroulement du test	21
6.5.1.10.3	Validation du test	21
6.5.1.11	Exigences	21
6.5.2	apTvAOTestimationMultiSpectral	21
6.5.2.1	Objectif	21
6.5.2.2	Description	21
6.5.2.3	Liste des données d'entrées	21
6.5.2.4	Liste des produits de sortie	22
6.5.2.5	Prérequis	22
6.5.2.6	Durée attendue	22
6.5.2.7	Epsilon utilisé sur la non regression	22
6.5.2.8	Vérifications à effectuer	22
6.5.2.9	Mise en oeuvre du test	22
6.5.2.10	Journal d'essai de la recette	22
6.5.2.10.1	Notes sur le déroulement du test	22
6.5.2.10.2	Conclusion du déroulement du test	22
6.5.2.10.3	Validation du test	22
6.5.2.11	Exigences	23
6.5.3	apTvAOTestimation_MultiTemporalNominal	23
6.5.3.1	Objectif	23
6.5.3.2	Description	23
6.5.3.3	Liste des données d'entrées	23
6.5.3.4	Liste des produits de sortie	24
6.5.3.5	Prérequis	24
6.5.3.6	Durée attendue	24
6.5.3.7	Epsilon utilisé sur la non regression	24
6.5.3.8	Vérifications à effectuer	24
6.5.3.9	Mise en oeuvre du test	24
6.5.3.10	Journal d'essai de la recette	24
6.5.3.10.1	Notes sur le déroulement du test	24
6.5.3.10.2	Conclusion du déroulement du test	24
6.5.3.10.3	Validation du test	24
6.5.3.11	Exigences	25
6.5.4	apTvAOTestimationMultiTemporalCAMS	25
6.5.4.1	Objectif	25
6.5.4.2	Description	25
6.5.4.3	Liste des données d'entrées	25
6.5.4.4	Liste des produits de sortie	26
6.5.4.5	Prérequis	26
6.5.4.6	Durée attendue	26
6.5.4.7	Epsilon utilisé sur la non regression	26
6.5.4.8	Vérifications à effectuer	26
6.5.4.9	Mise en oeuvre du test	26
6.5.4.10	Journal d'essai de la recette	26
6.5.4.10.1	Notes sur le déroulement du test	26

	6.5.4.10.2	Conclusion du déroulement du test	26
	6.5.4.10.3	Validation du test	27
	6.5.4.11	Exigences	27
6.5.5	apTvAOTEstimation_SpectroTemporalNominal		27
	6.5.5.1	Objectif	27
	6.5.5.2	Description	27
	6.5.5.3	Liste des données d'entrées	27
	6.5.5.4	Liste des produits de sortie	28
	6.5.5.5	Prérequis	28
	6.5.5.6	Durée attendue	28
	6.5.5.7	Epsilon utilisé sur la non regression	28
	6.5.5.8	Vérifications à effectuer	28
	6.5.5.9	Mise en oeuvre du test	28
	6.5.5.10	Journal d'essai de la recette	28
	6.5.5.10.1	Notes sur le déroulement du test	28
	6.5.5.10.2	Conclusion du déroulement du test	28
	6.5.5.10.3	Validation du test	29
	6.5.5.11	Exigences	29
6.5.6	apTvAOTMean_SpectroTemporalNominal		29
	6.5.6.1	Objectif	29
	6.5.6.2	Description	29
	6.5.6.3	Liste des données d'entrées	29
	6.5.6.4	Liste des produits de sortie	30
	6.5.6.5	Prérequis	30
	6.5.6.6	Durée attendue	30
	6.5.6.7	Epsilon utilisé sur la non regression	30
	6.5.6.8	Vérifications à effectuer	30
	6.5.6.9	Mise en oeuvre du test	30
	6.5.6.10	Journal d'essai de la recette	30
	6.5.6.10.1	Notes sur le déroulement du test	30
	6.5.6.10.2	Conclusion du déroulement du test	30
	6.5.6.10.3	Validation du test	31
	6.5.6.11	Exigences	31
6.5.7	apTvBinaryConcatenate		31
	6.5.7.1	Objectif	31
	6.5.7.2	Description	31
	6.5.7.3	Liste des données d'entrées	31
	6.5.7.4	Liste des produits de sortie	31
	6.5.7.5	Prérequis	31
	6.5.7.6	Durée attendue	31
	6.5.7.7	Epsilon utilisé sur la non regression	31
	6.5.7.8	Vérifications à effectuer	32
	6.5.7.9	Mise en oeuvre du test	32
	6.5.7.10	Journal d'essai de la recette	32
	6.5.7.10.1	Notes sur le déroulement du test	32
	6.5.7.10.2	Conclusion du déroulement du test	32
	6.5.7.10.3	Validation du test	32
	6.5.7.11	Exigences	32
6.5.8	apTvBinaryToVector		32
	6.5.8.1	Objectif	32
	6.5.8.2	Description	32
	6.5.8.3	Liste des données d'entrées	33
	6.5.8.4	Liste des produits de sortie	33
	6.5.8.5	Prérequis	33

6.5.8.6	Durée attendue	33
6.5.8.7	Epsilon utilisé sur la non regression	33
6.5.8.8	Vérifications à effectuer	33
6.5.8.9	Mise en oeuvre du test	33
6.5.8.10	Journal d'essai de la recette	33
6.5.8.10.1	Notes sur le déroulement du test	33
6.5.8.10.2	Conclusion du déroulement du test	33
6.5.8.10.3	Validation du test	34
6.5.8.11	Exigences	34
6.5.9	apTvCamsCompute	34
6.5.9.1	Objectif	34
6.5.9.2	Description	34
6.5.9.3	Liste des données d'entrées	34
6.5.9.4	Liste des produits de sortie	34
6.5.9.5	Prérequis	34
6.5.9.6	Durée attendue	34
6.5.9.7	Epsilon utilisé sur la non regression	34
6.5.9.8	Vérifications à effectuer	35
6.5.9.9	Mise en oeuvre du test	35
6.5.9.10	Journal d'essai de la recette	35
6.5.9.10.1	Notes sur le déroulement du test	35
6.5.9.10.2	Conclusion du déroulement du test	35
6.5.9.10.3	Validation du test	35
6.5.9.11	Exigences	35
6.5.10	apTvCirrusCorrectionApply	35
6.5.10.1	Objectif	35
6.5.10.2	Description	35
6.5.10.3	Liste des données d'entrées	36
6.5.10.4	Liste des produits de sortie	36
6.5.10.5	Prérequis	36
6.5.10.6	Durée attendue	36
6.5.10.7	Epsilon utilisé sur la non regression	36
6.5.10.8	Vérifications à effectuer	37
6.5.10.9	Mise en oeuvre du test	37
6.5.10.10	Journal d'essai de la recette	37
6.5.10.10.1	Notes sur le déroulement du test	37
6.5.10.10.2	Conclusion du déroulement du test	37
6.5.10.10.3	Validation du test	37
6.5.10.11	Exigences	37
6.5.11	apTvCirrusFlag	37
6.5.11.1	Objectif	37
6.5.11.2	Description	37
6.5.11.3	Liste des données d'entrées	38
6.5.11.4	Liste des produits de sortie	38
6.5.11.5	Prérequis	38
6.5.11.6	Durée attendue	38
6.5.11.7	Epsilon utilisé sur la non regression	38
6.5.11.8	Vérifications à effectuer	38
6.5.11.9	Mise en oeuvre du test	38
6.5.11.10	Journal d'essai de la recette	38
6.5.11.10.1	Notes sur le déroulement du test	38
6.5.11.10.2	Conclusion du déroulement du test	39
6.5.11.10.3	Validation du test	39
6.5.11.11	Exigences	39

6.5.12	apTvCirrusMask	39
6.5.12.1	Objectif	39
6.5.12.2	Description	39
6.5.12.3	Liste des données d'entrées	39
6.5.12.4	Liste des produits de sortie	40
6.5.12.5	Prérequis	40
6.5.12.6	Durée attendue	40
6.5.12.7	Epsilon utilisé sur la non regression	40
6.5.12.8	Vérifications à effectuer	40
6.5.12.9	Mise en oeuvre du test	40
6.5.12.10	Journal d'essai de la recette	40
6.5.12.10.1	Notes sur le déroulement du test	40
6.5.12.10.2	Conclusion du déroulement du test	40
6.5.12.10.3	Validation du test	40
6.5.12.11	Exigences	41
6.5.13	apTvCloudReflectance	41
6.5.13.1	Objectif	41
6.5.13.2	Description	41
6.5.13.3	Liste des données d'entrées	41
6.5.13.4	Liste des produits de sortie	42
6.5.13.5	Prérequis	42
6.5.13.6	Durée attendue	42
6.5.13.7	Epsilon utilisé sur la non regression	42
6.5.13.8	Vérifications à effectuer	42
6.5.13.9	Mise en oeuvre du test	42
6.5.13.10	Journal d'essai de la recette	42
6.5.13.10.1	Notes sur le déroulement du test	42
6.5.13.10.2	Conclusion du déroulement du test	42
6.5.13.10.3	Validation du test	42
6.5.13.11	Exigences	43
6.5.14	apTvCloudMaskUndersampling	43
6.5.14.1	Objectif	43
6.5.14.2	Description	43
6.5.14.3	Liste des données d'entrées	43
6.5.14.4	Liste des produits de sortie	43
6.5.14.5	Prérequis	43
6.5.14.6	Durée attendue	43
6.5.14.7	Epsilon utilisé sur la non regression	44
6.5.14.8	Vérifications à effectuer	44
6.5.14.9	Mise en oeuvre du test	44
6.5.14.10	Journal d'essai de la recette	44
6.5.14.10.1	Notes sur le déroulement du test	44
6.5.14.10.2	Conclusion du déroulement du test	44
6.5.14.10.3	Validation du test	44
6.5.14.11	Exigences	44
6.5.15	apTvCloudReflectance	45
6.5.15.1	Objectif	45
6.5.15.2	Description	45
6.5.15.3	Liste des données d'entrées	45
6.5.15.4	Liste des produits de sortie	45
6.5.15.5	Prérequis	46
6.5.15.6	Durée attendue	46
6.5.15.7	Epsilon utilisé sur la non regression	46
6.5.15.8	Vérifications à effectuer	46

6.5.15.9	Mise en oeuvre du test	46
6.5.15.10	Journal d'essai de la recette	46
6.5.15.10.1	Notes sur le déroulement du test	46
6.5.15.10.2	Conclusion du déroulement du test	46
6.5.15.10.3	Validation du test	46
6.5.15.11	Exigences	46
6.5.16	apTvCloudReflectance	47
6.5.16.1	Objectif	47
6.5.16.2	Description	47
6.5.16.3	Liste des données d'entrées	47
6.5.16.4	Liste des produits de sortie	48
6.5.16.5	Prérequis	48
6.5.16.6	Durée attendue	48
6.5.16.7	Epsilon utilisé sur la non regression	49
6.5.16.8	Vérifications à effectuer	49
6.5.16.9	Mise en oeuvre du test	49
6.5.16.10	Journal d'essai de la recette	49
6.5.16.10.1	Notes sur le déroulement du test	49
6.5.16.10.2	Conclusion du déroulement du test	49
6.5.16.10.3	Validation du test	49
6.5.16.11	Exigences	49
6.5.17	apTvCloudShadowWithAltitudeFilterInit	50
6.5.17.1	Objectif	50
6.5.17.2	Description	50
6.5.17.3	Liste des données d'entrées	50
6.5.17.4	Liste des produits de sortie	51
6.5.17.5	Prérequis	51
6.5.17.6	Durée attendue	51
6.5.17.7	Epsilon utilisé sur la non regression	51
6.5.17.8	Vérifications à effectuer	51
6.5.17.9	Mise en oeuvre du test	51
6.5.17.10	Journal d'essai de la recette	51
6.5.17.10.1	Notes sur le déroulement du test	51
6.5.17.10.2	Conclusion du déroulement du test	51
6.5.17.10.3	Validation du test	52
6.5.17.11	Exigences	52
6.5.18	apTvCloudShadowWithoutAltitudeFilterNominal	52
6.5.18.1	Objectif	52
6.5.18.2	Description	52
6.5.18.3	Liste des données d'entrées	52
6.5.18.4	Liste des produits de sortie	53
6.5.18.5	Prérequis	53
6.5.18.6	Durée attendue	53
6.5.18.7	Epsilon utilisé sur la non regression	53
6.5.18.8	Vérifications à effectuer	54
6.5.18.9	Mise en oeuvre du test	54
6.5.18.10	Journal d'essai de la recette	54
6.5.18.10.1	Notes sur le déroulement du test	54
6.5.18.10.2	Conclusion du déroulement du test	54
6.5.18.10.3	Validation du test	54
6.5.18.11	Exigences	54
6.5.19	apTvCloudShadVar	54
6.5.19.1	Objectif	54
6.5.19.2	Description	55

6.5.19.3	Liste des données d'entrées	55
6.5.19.4	Liste des produits de sortie	56
6.5.19.5	Prérequis	56
6.5.19.6	Durée attendue	56
6.5.19.7	Epsilon utilisé sur la non regression	56
6.5.19.8	Vérifications à effectuer	56
6.5.19.9	Mise en oeuvre du test	56
6.5.19.10	Journal d'essai de la recette	56
6.5.19.10.1	Notes sur le déroulement du test	56
6.5.19.10.2	Conclusion du déroulement du test	56
6.5.19.10.3	Validation du test	56
6.5.19.11	Exigences	57
6.5.20	apTvCloudSnowCorrection	57
6.5.20.1	Objectif	57
6.5.20.2	Description	57
6.5.20.3	Liste des données d'entrées	57
6.5.20.4	Liste des produits de sortie	57
6.5.20.5	Prérequis	57
6.5.20.6	Durée attendue	57
6.5.20.7	Epsilon utilisé sur la non regression	58
6.5.20.8	Vérifications à effectuer	58
6.5.20.9	Mise en oeuvre du test	58
6.5.20.10	Journal d'essai de la recette	58
6.5.20.10.1	Notes sur le déroulement du test	58
6.5.20.10.2	Conclusion du déroulement du test	58
6.5.20.10.3	Validation du test	58
6.5.20.11	Exigences	58
6.5.21	apTvComposite	58
6.5.21.1	Objectif	58
6.5.21.2	Description	59
6.5.21.3	Liste des données d'entrées	59
6.5.21.4	Liste des produits de sortie	59
6.5.21.5	Prérequis	60
6.5.21.6	Durée attendue	60
6.5.21.7	Epsilon utilisé sur la non regression	60
6.5.21.8	Vérifications à effectuer	60
6.5.21.9	Mise en oeuvre du test	60
6.5.21.10	Journal d'essai de la recette	60
6.5.21.10.1	Notes sur le déroulement du test	60
6.5.21.10.2	Conclusion du déroulement du test	60
6.5.21.10.3	Validation du test	60
6.5.21.11	Exigences	60
6.5.22	apTvComputeRhoEnv	61
6.5.22.1	Objectif	61
6.5.22.2	Description	61
6.5.22.3	Liste des données d'entrées	61
6.5.22.4	Liste des produits de sortie	61
6.5.22.5	Prérequis	61
6.5.22.6	Durée attendue	61
6.5.22.7	Epsilon utilisé sur la non regression	61
6.5.22.8	Vérifications à effectuer	61
6.5.22.9	Mise en oeuvre du test	61
6.5.22.10	Journal d'essai de la recette	62
6.5.22.10.1	Notes sur le déroulement du test	62

6.5.22.10.2	Conclusion du déroulement du test	62
6.5.22.10.3	Validation du test	62
6.5.22.11	Exigences	62
6.5.23	apTvComputeTransmission	62
6.5.23.1	Objectif	62
6.5.23.2	Description	62
6.5.23.3	Liste des données d'entrées	62
6.5.23.4	Liste des produits de sortie	63
6.5.23.5	Prérequis	63
6.5.23.6	Durée attendue	63
6.5.23.7	Epsilon utilisé sur la non regression	63
6.5.23.8	Vérifications à effectuer	63
6.5.23.9	Mise en oeuvre du test	63
6.5.23.10	Journal d'essai de la recette	63
6.5.23.10.1	Notes sur le déroulement du test	63
6.5.23.10.2	Conclusion du déroulement du test	63
6.5.23.10.3	Validation du test	63
6.5.23.11	Exigences	64
6.5.24	apTvConcatenatePerZone	64
6.5.24.1	Objectif	64
6.5.24.2	Description	64
6.5.24.3	Liste des données d'entrées	64
6.5.24.4	Liste des produits de sortie	64
6.5.24.5	Prérequis	64
6.5.24.6	Durée attendue	64
6.5.24.7	Epsilon utilisé sur la non regression	64
6.5.24.8	Vérifications à effectuer	64
6.5.24.9	Mise en oeuvre du test	65
6.5.24.10	Journal d'essai de la recette	65
6.5.24.10.1	Notes sur le déroulement du test	65
6.5.24.10.2	Conclusion du déroulement du test	65
6.5.24.10.3	Validation du test	65
6.5.24.11	Exigences	65
6.5.25	apTvDarkestPixelAOTNominal	65
6.5.25.1	Objectif	65
6.5.25.2	Description	65
6.5.25.3	Liste des données d'entrées	65
6.5.25.4	Liste des produits de sortie	66
6.5.25.5	Prérequis	66
6.5.25.6	Durée attendue	66
6.5.25.7	Epsilon utilisé sur la non regression	66
6.5.25.8	Vérifications à effectuer	66
6.5.25.9	Mise en oeuvre du test	66
6.5.25.10	Journal d'essai de la recette	66
6.5.25.10.1	Notes sur le déroulement du test	66
6.5.25.10.2	Conclusion du déroulement du test	67
6.5.25.10.3	Validation du test	67
6.5.25.11	Exigences	67
6.5.26	apTvDirectionalCorrectionROY	67
6.5.26.1	Objectif	67
6.5.26.2	Description	67
6.5.26.3	Liste des données d'entrées	67
6.5.26.4	Liste des produits de sortie	67
6.5.26.5	Prérequis	67

6.5.26.6	Durée attendue	67
6.5.26.7	Epsilon utilisé sur la non regression	68
6.5.26.8	Vérifications à effectuer	68
6.5.26.9	Mise en oeuvre du test	68
6.5.26.10	Journal d'essai de la recette	68
6.5.26.10.1	Notes sur le déroulement du test	68
6.5.26.10.2	Conclusion du déroulement du test	68
6.5.26.10.3	Validation du test	68
6.5.26.11	Exigences	68
6.5.27	apTvDirectionalCorrectionLUT	68
6.5.27.1	Objectif	68
6.5.27.2	Description	69
6.5.27.3	Liste des données d'entrées	69
6.5.27.4	Liste des produits de sortie	69
6.5.27.5	Prérequis	69
6.5.27.6	Durée attendue	69
6.5.27.7	Epsilon utilisé sur la non regression	69
6.5.27.8	Vérifications à effectuer	69
6.5.27.9	Mise en oeuvre du test	69
6.5.27.10	Journal d'essai de la recette	69
6.5.27.10.1	Notes sur le déroulement du test	69
6.5.27.10.2	Conclusion du déroulement du test	70
6.5.27.10.3	Validation du test	70
6.5.27.11	Exigences	70
6.5.28	apTvDispatchZoneToVector	70
6.5.28.1	Objectif	70
6.5.28.2	Description	70
6.5.28.3	Liste des données d'entrées	70
6.5.28.4	Liste des produits de sortie	70
6.5.28.5	Prérequis	70
6.5.28.6	Durée attendue	71
6.5.28.7	Epsilon utilisé sur la non regression	71
6.5.28.8	Vérifications à effectuer	71
6.5.28.9	Mise en oeuvre du test	71
6.5.28.10	Journal d'essai de la recette	71
6.5.28.10.1	Notes sur le déroulement du test	71
6.5.28.10.2	Conclusion du déroulement du test	71
6.5.28.10.3	Validation du test	71
6.5.28.11	Exigences	71
6.5.29	apTvDispatchZoneToVectorWeight	71
6.5.29.1	Objectif	71
6.5.29.2	Description	72
6.5.29.3	Liste des données d'entrées	72
6.5.29.4	Liste des produits de sortie	72
6.5.29.5	Prérequis	72
6.5.29.6	Durée attendue	72
6.5.29.7	Epsilon utilisé sur la non regression	72
6.5.29.8	Vérifications à effectuer	72
6.5.29.9	Mise en oeuvre du test	72
6.5.29.10	Journal d'essai de la recette	73
6.5.29.10.1	Notes sur le déroulement du test	73
6.5.29.10.2	Conclusion du déroulement du test	73
6.5.29.10.3	Validation du test	73
6.5.29.11	Exigences	73

6.5.30	apTvDTMProcessing	73
6.5.30.1	Objectif	73
6.5.30.2	Description	73
6.5.30.3	Liste des données d'entrées	73
6.5.30.4	Liste des produits de sortie	74
6.5.30.5	Prérequis	74
6.5.30.6	Durée attendue	74
6.5.30.7	Epsilon utilisé sur la non regression	74
6.5.30.8	Vérifications à effectuer	74
6.5.30.9	Mise en oeuvre du test	74
6.5.30.10	Journal d'essai de la recette	74
6.5.30.10.1	Notes sur le déroulement du test	74
6.5.30.10.2	Conclusion du déroulement du test	74
6.5.30.10.3	Validation du test	74
6.5.30.11	Exigences	75
6.5.31	apTvEnvCorrection	75
6.5.31.1	Objectif	75
6.5.31.2	Description	75
6.5.31.3	Liste des données d'entrées	75
6.5.31.4	Liste des produits de sortie	75
6.5.31.5	Prérequis	75
6.5.31.6	Durée attendue	76
6.5.31.7	Epsilon utilisé sur la non regression	76
6.5.31.8	Vérifications à effectuer	76
6.5.31.9	Mise en oeuvre du test	76
6.5.31.10	Journal d'essai de la recette	76
6.5.31.10.1	Notes sur le déroulement du test	76
6.5.31.10.2	Conclusion du déroulement du test	76
6.5.31.10.3	Validation du test	76
6.5.31.11	Exigences	76
6.5.32	apTvExtractLutChannel	76
6.5.32.1	Objectif	76
6.5.32.2	Description	77
6.5.32.3	Liste des données d'entrées	77
6.5.32.4	Liste des produits de sortie	77
6.5.32.5	Prérequis	77
6.5.32.6	Durée attendue	77
6.5.32.7	Epsilon utilisé sur la non regression	77
6.5.32.8	Vérifications à effectuer	77
6.5.32.9	Mise en oeuvre du test	77
6.5.32.10	Journal d'essai de la recette	77
6.5.32.10.1	Notes sur le déroulement du test	77
6.5.32.10.2	Conclusion du déroulement du test	78
6.5.32.10.3	Validation du test	78
6.5.32.11	Exigences	78
6.5.33	apTvGammaCompute	78
6.5.33.1	Objectif	78
6.5.33.2	Description	78
6.5.33.3	Liste des données d'entrées	78
6.5.33.4	Liste des produits de sortie	79
6.5.33.5	Prérequis	79
6.5.33.6	Durée attendue	79
6.5.33.7	Epsilon utilisé sur la non regression	79
6.5.33.8	Vérifications à effectuer	79

6.5.33.9	Mise en oeuvre du test	79
6.5.33.10	Journal d'essai de la recette	79
6.5.33.10.1	Notes sur le déroulement du test	79
6.5.33.10.2	Conclusion du déroulement du test	79
6.5.33.10.3	Validation du test	79
6.5.33.11	Exigences	80
6.5.34	apTvGapFilling	80
6.5.34.1	Objectif	80
6.5.34.2	Description	80
6.5.34.3	Liste des données d'entrées	80
6.5.34.4	Liste des produits de sortie	80
6.5.34.5	Prérequis	80
6.5.34.6	Durée attendue	80
6.5.34.7	Epsilon utilisé sur la non regression	81
6.5.34.8	Vérifications à effectuer	81
6.5.34.9	Mise en oeuvre du test	81
6.5.34.10	Journal d'essai de la recette	81
6.5.34.10.1	Notes sur le déroulement du test	81
6.5.34.10.2	Conclusion du déroulement du test	81
6.5.34.10.3	Validation du test	81
6.5.34.11	Exigences	81
6.5.35	apTvImageListWriter	81
6.5.35.1	Objectif	81
6.5.35.2	Description	82
6.5.35.3	Liste des données d'entrées	82
6.5.35.4	Liste des produits de sortie	82
6.5.35.5	Prérequis	82
6.5.35.6	Durée attendue	82
6.5.35.7	Epsilon utilisé sur la non regression	82
6.5.35.8	Vérifications à effectuer	82
6.5.35.9	Mise en oeuvre du test	82
6.5.35.10	Journal d'essai de la recette	82
6.5.35.10.1	Notes sur le déroulement du test	82
6.5.35.10.2	Conclusion du déroulement du test	83
6.5.35.10.3	Validation du test	83
6.5.35.11	Exigences	83
6.5.36	apTvIncidenceAngle	83
6.5.36.1	Objectif	83
6.5.36.2	Description	83
6.5.36.3	Liste des données d'entrées	83
6.5.36.4	Liste des produits de sortie	83
6.5.36.5	Prérequis	83
6.5.36.6	Durée attendue	84
6.5.36.7	Epsilon utilisé sur la non regression	84
6.5.36.8	Vérifications à effectuer	84
6.5.36.9	Mise en oeuvre du test	84
6.5.36.10	Journal d'essai de la recette	84
6.5.36.10.1	Notes sur le déroulement du test	84
6.5.36.10.2	Conclusion du déroulement du test	84
6.5.36.10.3	Validation du test	84
6.5.36.11	Exigences	84
6.5.37	apTvInterpolateForEnvCor	84
6.5.37.1	Objectif	84
6.5.37.2	Description	85

6.5.37.3	Liste des données d'entrées	85
6.5.37.4	Liste des produits de sortie	85
6.5.37.5	Prérequis	85
6.5.37.6	Durée attendue	85
6.5.37.7	Epsilon utilisé sur la non regression	85
6.5.37.8	Vérifications à effectuer	85
6.5.37.9	Mise en oeuvre du test	86
6.5.37.10	Journal d'essai de la recette	86
6.5.37.10.1	Notes sur le déroulement du test	86
6.5.37.10.2	Conclusion du déroulement du test	86
6.5.37.10.3	Validation du test	86
6.5.37.11	Exigences	86
6.5.38	apTvMultiplyByScalar	86
6.5.38.1	Objectif	86
6.5.38.2	Description	86
6.5.38.3	Liste des données d'entrées	86
6.5.38.4	Liste des produits de sortie	87
6.5.38.5	Prérequis	87
6.5.38.6	Durée attendue	87
6.5.38.7	Epsilon utilisé sur la non regression	87
6.5.38.8	Vérifications à effectuer	87
6.5.38.9	Mise en oeuvre du test	87
6.5.38.10	Journal d'essai de la recette	87
6.5.38.10.1	Notes sur le déroulement du test	87
6.5.38.10.2	Conclusion du déroulement du test	87
6.5.38.10.3	Validation du test	87
6.5.38.11	Exigences	88
6.5.39	apTvNewCamsCompute	88
6.5.39.1	Objectif	88
6.5.39.2	Description	88
6.5.39.3	Liste des données d'entrées	88
6.5.39.4	Liste des produits de sortie	88
6.5.39.5	Prérequis	88
6.5.39.6	Durée attendue	88
6.5.39.7	Epsilon utilisé sur la non regression	88
6.5.39.8	Vérifications à effectuer	88
6.5.39.9	Mise en oeuvre du test	89
6.5.39.10	Journal d'essai de la recette	89
6.5.39.10.1	Notes sur le déroulement du test	89
6.5.39.10.2	Conclusion du déroulement du test	89
6.5.39.10.3	Validation du test	89
6.5.39.11	Exigences	89
6.5.40	apTvNewCamsMixedCompute	89
6.5.40.1	Objectif	89
6.5.40.2	Description	89
6.5.40.3	Liste des données d'entrées	89
6.5.40.4	Liste des produits de sortie	90
6.5.40.5	Prérequis	90
6.5.40.6	Durée attendue	90
6.5.40.7	Epsilon utilisé sur la non regression	90
6.5.40.8	Vérifications à effectuer	90
6.5.40.9	Mise en oeuvre du test	90
6.5.40.10	Journal d'essai de la recette	90
6.5.40.10.1	Notes sur le déroulement du test	90

	6.5.40.10.2 Conclusion du déroulement du test	90
	6.5.40.10.3 Validation du test	90
	6.5.40.11 Exigences	91
6.5.41	apTvOneBandEqualThreshold	91
	6.5.41.1 Objectif	91
	6.5.41.2 Description	91
	6.5.41.3 Liste des données d'entrées	91
	6.5.41.4 Liste des produits de sortie	91
	6.5.41.5 Prérequis	91
	6.5.41.6 Durée attendue	91
	6.5.41.7 Epsilon utilisé sur la non regression	91
	6.5.41.8 Vérifications à effectuer	91
	6.5.41.9 Mise en oeuvre du test	92
	6.5.41.10 Journal d'essai de la recette	92
	6.5.41.10.1 Notes sur le déroulement du test	92
	6.5.41.10.2 Conclusion du déroulement du test	92
	6.5.41.10.3 Validation du test	92
	6.5.41.11 Exigences	92
6.5.42	apTvOzoneExtractor	92
	6.5.42.1 Objectif	92
	6.5.42.2 Description	92
	6.5.42.3 Liste des données d'entrées	92
	6.5.42.4 Liste des produits de sortie	93
	6.5.42.5 Prérequis	93
	6.5.42.6 Durée attendue	93
	6.5.42.7 Epsilon utilisé sur la non regression	93
	6.5.42.8 Vérifications à effectuer	93
	6.5.42.9 Mise en oeuvre du test	93
	6.5.42.10 Journal d'essai de la recette	93
	6.5.42.10.1 Notes sur le déroulement du test	93
	6.5.42.10.2 Conclusion du déroulement du test	93
	6.5.42.10.3 Validation du test	93
	6.5.42.11 Exigences	94
6.5.43	apTvPressure	94
	6.5.43.1 Objectif	94
	6.5.43.2 Description	94
	6.5.43.3 Liste des données d'entrées	94
	6.5.43.4 Liste des produits de sortie	94
	6.5.43.5 Prérequis	94
	6.5.43.6 Durée attendue	94
	6.5.43.7 Epsilon utilisé sur la non regression	94
	6.5.43.8 Vérifications à effectuer	94
	6.5.43.9 Mise en oeuvre du test	95
	6.5.43.10 Journal d'essai de la recette	95
	6.5.43.10.1 Notes sur le déroulement du test	95
	6.5.43.10.2 Conclusion du déroulement du test	95
	6.5.43.10.3 Validation du test	95
	6.5.43.11 Exigences	95
6.5.44	apTvRayleighCorrection	95
	6.5.44.1 Objectif	95
	6.5.44.2 Description	95
	6.5.44.3 Liste des données d'entrées	95
	6.5.44.4 Liste des produits de sortie	96
	6.5.44.5 Prérequis	96

6.5.44.6	Durée attendue	96
6.5.44.7	Epsilon utilisé sur la non regression	96
6.5.44.8	Vérifications à effectuer	96
6.5.44.9	Mise en oeuvre du test	96
6.5.44.10	Journal d'essai de la recette	96
6.5.44.10.1	Notes sur le déroulement du test	96
6.5.44.10.2	Conclusion du déroulement du test	96
6.5.44.10.3	Validation du test	96
6.5.44.11	Exigences	97
6.5.45	apTvReduceLutDirCorr	97
6.5.45.1	Objectif	97
6.5.45.2	Description	97
6.5.45.3	Liste des données d'entrées	97
6.5.45.4	Liste des produits de sortie	97
6.5.45.5	Prérequis	98
6.5.45.6	Durée attendue	98
6.5.45.7	Epsilon utilisé sur la non regression	98
6.5.45.8	Vérifications à effectuer	98
6.5.45.9	Mise en oeuvre du test	98
6.5.45.10	Journal d'essai de la recette	98
6.5.45.10.1	Notes sur le déroulement du test	98
6.5.45.10.2	Conclusion du déroulement du test	98
6.5.45.10.3	Validation du test	98
6.5.45.11	Exigences	98
6.5.46	apTvReduceLut	99
6.5.46.1	Objectif	99
6.5.46.2	Description	99
6.5.46.3	Liste des données d'entrées	99
6.5.46.4	Liste des produits de sortie	99
6.5.46.5	Prérequis	99
6.5.46.6	Durée attendue	99
6.5.46.7	Epsilon utilisé sur la non regression	99
6.5.46.8	Vérifications à effectuer	99
6.5.46.9	Mise en oeuvre du test	100
6.5.46.10	Journal d'essai de la recette	100
6.5.46.10.1	Notes sur le déroulement du test	100
6.5.46.10.2	Conclusion du déroulement du test	100
6.5.46.10.3	Validation du test	100
6.5.46.11	Exigences	100
6.5.47	apTvResampling	100
6.5.47.1	Objectif	100
6.5.47.2	Description	100
6.5.47.3	Liste des données d'entrées	100
6.5.47.4	Liste des produits de sortie	101
6.5.47.5	Prérequis	101
6.5.47.6	Durée attendue	101
6.5.47.7	Epsilon utilisé sur la non regression	101
6.5.47.8	Vérifications à effectuer	101
6.5.47.9	Mise en oeuvre du test	101
6.5.47.10	Journal d'essai de la recette	101
6.5.47.10.1	Notes sur le déroulement du test	101
6.5.47.10.2	Conclusion du déroulement du test	101
6.5.47.10.3	Validation du test	101
6.5.47.11	Exigences	102

6.5.48	apTvRescaleIntensity	102
6.5.48.1	Objectif	102
6.5.48.2	Description	102
6.5.48.3	Liste des données d'entrées	102
6.5.48.4	Liste des produits de sortie	102
6.5.48.5	Prérequis	102
6.5.48.6	Durée attendue	102
6.5.48.7	Epsilon utilisé sur la non regression	102
6.5.48.8	Vérifications à effectuer	102
6.5.48.9	Mise en oeuvre du test	103
6.5.48.10	Journal d'essai de la recette	103
6.5.48.10.1	Notes sur le déroulement du test	103
6.5.48.10.2	Conclusion du déroulement du test	103
6.5.48.10.3	Validation du test	103
6.5.48.11	Exigences	103
6.5.49	apTvRoundImage	103
6.5.49.1	Objectif	103
6.5.49.2	Description	103
6.5.49.3	Liste des données d'entrées	103
6.5.49.4	Liste des produits de sortie	104
6.5.49.5	Prérequis	104
6.5.49.6	Durée attendue	104
6.5.49.7	Epsilon utilisé sur la non regression	104
6.5.49.8	Vérifications à effectuer	104
6.5.49.9	Mise en oeuvre du test	104
6.5.49.10	Journal d'essai de la recette	104
6.5.49.10.1	Notes sur le déroulement du test	104
6.5.49.10.2	Conclusion du déroulement du test	104
6.5.49.10.3	Validation du test	104
6.5.49.11	Exigences	105
6.5.50	apTvRoundVectorImage	105
6.5.50.1	Objectif	105
6.5.50.2	Description	105
6.5.50.3	Liste des données d'entrées	105
6.5.50.4	Liste des produits de sortie	105
6.5.50.5	Prérequis	105
6.5.50.6	Durée attendue	105
6.5.50.7	Epsilon utilisé sur la non regression	105
6.5.50.8	Vérifications à effectuer	105
6.5.50.9	Mise en oeuvre du test	106
6.5.50.10	Journal d'essai de la recette	106
6.5.50.10.1	Notes sur le déroulement du test	106
6.5.50.10.2	Conclusion du déroulement du test	106
6.5.50.10.3	Validation du test	106
6.5.50.11	Exigences	106
6.5.51	apTvScatteringCorrection	106
6.5.51.1	Objectif	106
6.5.51.2	Description	106
6.5.51.3	Liste des données d'entrées	106
6.5.51.4	Liste des produits de sortie	107
6.5.51.5	Prérequis	107
6.5.51.6	Durée attendue	107
6.5.51.7	Epsilon utilisé sur la non regression	107
6.5.51.8	Vérifications à effectuer	107

6.5.51.9	Mise en oeuvre du test	107
6.5.51.10	Journal d'essai de la recette	107
6.5.51.10.1	Notes sur le déroulement du test	107
6.5.51.10.2	Conclusion du déroulement du test	107
6.5.51.10.3	Validation du test	107
6.5.51.11	Exigences	108
6.5.52	apTvSlopeCorrection	108
6.5.52.1	Objectif	108
6.5.52.2	Description	108
6.5.52.3	Liste des données d'entrées	108
6.5.52.4	Liste des produits de sortie	109
6.5.52.5	Prérequis	109
6.5.52.6	Durée attendue	109
6.5.52.7	Epsilon utilisé sur la non regression	109
6.5.52.8	Vérifications à effectuer	109
6.5.52.9	Mise en oeuvre du test	109
6.5.52.10	Journal d'essai de la recette	109
6.5.52.10.1	Notes sur le déroulement du test	109
6.5.52.10.2	Conclusion du déroulement du test	109
6.5.52.10.3	Validation du test	109
6.5.52.11	Exigences	110
6.5.53	apTvSmoothingBis	110
6.5.53.1	Objectif	110
6.5.53.2	Description	110
6.5.53.3	Liste des données d'entrées	110
6.5.53.4	Liste des produits de sortie	110
6.5.53.5	Prérequis	110
6.5.53.6	Durée attendue	110
6.5.53.7	Epsilon utilisé sur la non regression	110
6.5.53.8	Vérifications à effectuer	110
6.5.53.9	Mise en oeuvre du test	111
6.5.53.10	Journal d'essai de la recette	111
6.5.53.10.1	Notes sur le déroulement du test	111
6.5.53.10.2	Conclusion du déroulement du test	111
6.5.53.10.3	Validation du test	111
6.5.53.11	Exigences	111
6.5.54	apTvSnowMask	111
6.5.54.1	Objectif	111
6.5.54.2	Description	111
6.5.54.3	Liste des données d'entrées	111
6.5.54.4	Liste des produits de sortie	112
6.5.54.5	Prérequis	112
6.5.54.6	Durée attendue	112
6.5.54.7	Epsilon utilisé sur la non regression	112
6.5.54.8	Vérifications à effectuer	112
6.5.54.9	Mise en oeuvre du test	112
6.5.54.10	Journal d'essai de la recette	112
6.5.54.10.1	Notes sur le déroulement du test	112
6.5.54.10.2	Conclusion du déroulement du test	112
6.5.54.10.3	Validation du test	113
6.5.54.11	Exigences	113
6.5.55	apTvSnowMaskVenus	113
6.5.55.1	Objectif	113
6.5.55.2	Description	113

6.5.55.3	Liste des données d'entrées	113
6.5.55.4	Liste des produits de sortie	113
6.5.55.5	Prérequis	113
6.5.55.6	Durée attendue	113
6.5.55.7	Epsilon utilisé sur la non regression	113
6.5.55.8	Vérifications à effectuer	113
6.5.55.9	Mise en oeuvre du test	114
6.5.55.10	Journal d'essai de la recette	114
6.5.55.10.1	Notes sur le déroulement du test	114
6.5.55.10.2	Conclusion du déroulement du test	114
6.5.55.10.3	Validation du test	114
6.5.55.11	Exigences	114
6.5.56	apTvSolarAngle	114
6.5.56.1	Objectif	114
6.5.56.2	Description	114
6.5.56.3	Liste des données d'entrées	114
6.5.56.4	Liste des produits de sortie	115
6.5.56.5	Prérequis	115
6.5.56.6	Durée attendue	115
6.5.56.7	Epsilon utilisé sur la non regression	115
6.5.56.8	Vérifications à effectuer	115
6.5.56.9	Mise en oeuvre du test	115
6.5.56.10	Journal d'essai de la recette	115
6.5.56.10.1	Notes sur le déroulement du test	115
6.5.56.10.2	Conclusion du déroulement du test	115
6.5.56.10.3	Validation du test	115
6.5.56.11	Exigences	116
6.5.57	apTvStats	116
6.5.57.1	Objectif	116
6.5.57.2	Description	116
6.5.57.3	Liste des données d'entrées	116
6.5.57.4	Liste des produits de sortie	116
6.5.57.5	Prérequis	116
6.5.57.6	Durée attendue	116
6.5.57.7	Epsilon utilisé sur la non regression	116
6.5.57.8	Vérifications à effectuer	116
6.5.57.9	Mise en oeuvre du test	117
6.5.57.10	Journal d'essai de la recette	117
6.5.57.10.1	Notes sur le déroulement du test	117
6.5.57.10.2	Conclusion du déroulement du test	117
6.5.57.10.3	Validation du test	117
6.5.57.11	Exigences	117
6.5.58	apTvTOACorrection	117
6.5.58.1	Objectif	117
6.5.58.2	Description	117
6.5.58.3	Liste des données d'entrées	117
6.5.58.4	Liste des produits de sortie	118
6.5.58.5	Prérequis	118
6.5.58.6	Durée attendue	118
6.5.58.7	Epsilon utilisé sur la non regression	118
6.5.58.8	Vérifications à effectuer	118
6.5.58.9	Mise en oeuvre du test	118
6.5.58.10	Journal d'essai de la recette	118
6.5.58.10.1	Notes sur le déroulement du test	118

6.5.58.10.2	Conclusion du déroulement du test	118
6.5.58.10.3	Validation du test	119
6.5.58.11	Exigences	119
6.5.59	apTvWaterAmountGenerator	119
6.5.59.1	Objectif	119
6.5.59.2	Description	119
6.5.59.3	Liste des données d'entrées	119
6.5.59.4	Liste des produits de sortie	119
6.5.59.5	Prérequis	120
6.5.59.6	Durée attendue	120
6.5.59.7	Epsilon utilisé sur la non regression	120
6.5.59.8	Vérifications à effectuer	120
6.5.59.9	Mise en oeuvre du test	120
6.5.59.10	Journal d'essai de la recette	120
6.5.59.10.1	Notes sur le déroulement du test	120
6.5.59.10.2	Conclusion du déroulement du test	120
6.5.59.10.3	Validation du test	120
6.5.59.11	Exigences	120
6.5.60	apTvWaterMask	121
6.5.60.1	Objectif	121
6.5.60.2	Description	121
6.5.60.3	Liste des données d'entrées	121
6.5.60.4	Liste des produits de sortie	122
6.5.60.5	Prérequis	122
6.5.60.6	Durée attendue	122
6.5.60.7	Epsilon utilisé sur la non regression	122
6.5.60.8	Vérifications à effectuer	122
6.5.60.9	Mise en oeuvre du test	122
6.5.60.10	Journal d'essai de la recette	122
6.5.60.10.1	Notes sur le déroulement du test	122
6.5.60.10.2	Conclusion du déroulement du test	122
6.5.60.10.3	Validation du test	122
6.5.60.11	Exigences	123
6.5.61	apTvWaterVaporPostPro	123
6.5.61.1	Objectif	123
6.5.61.2	Description	123
6.5.61.3	Liste des données d'entrées	123
6.5.61.4	Liste des produits de sortie	123
6.5.61.5	Prérequis	124
6.5.61.6	Durée attendue	124
6.5.61.7	Epsilon utilisé sur la non regression	124
6.5.61.8	Vérifications à effectuer	124
6.5.61.9	Mise en oeuvre du test	124
6.5.61.10	Journal d'essai de la recette	124
6.5.61.10.1	Notes sur le déroulement du test	124
6.5.61.10.2	Conclusion du déroulement du test	124
6.5.61.10.3	Validation du test	124
6.5.61.11	Exigences	124
7	LES ESSAIS DE VALIDATION	125
7.1	Objectifs	125
7.2	Description	125
7.3	Formalisation des tests de validation	126
7.4	LISTE DES ESSAIS	127

7.4.1	pyTv-S2-L2NOMINAL-001-SENTINEL2-ALGO_CHAIN	129
7.4.1.1	Titre	129
7.4.1.2	Objectif	130
7.4.1.3	Description	130
7.4.1.4	Liste des données d'entrées	130
7.4.1.5	Liste des produits de sortie	131
7.4.1.6	Prérequis	131
7.4.1.7	Durée attendue	131
7.4.1.8	Epsilon utilisé sur la non regression	131
7.4.1.9	Vérifications à effectuer	131
7.4.1.10	Mise en oeuvre du test	131
7.4.1.11	Exigences	131
7.4.1.12	Journal d'essai de la recette	132
	7.4.1.12.1 Notes sur le déroulement du test	132
	7.4.1.12.2 Conclusion du déroulement du test	132
	7.4.1.12.3 Validation du test	132
7.4.2	pyTv-S2-L2INIT-011-SENTINEL2-MUSCATE-GENERAL_CHAIN	132
7.4.2.1	Titre	132
7.4.2.2	Objectif	132
7.4.2.3	Description	132
7.4.2.4	Liste des données d'entrées	133
7.4.2.5	Liste des produits de sortie	133
7.4.2.6	Prérequis	133
7.4.2.7	Durée attendue	133
7.4.2.8	Epsilon utilisé sur la non regression	133
7.4.2.9	Vérifications à effectuer	133
7.4.2.10	Mise en oeuvre du test	134
7.4.2.11	Exigences	134
7.4.2.12	Journal d'essai de la recette	134
	7.4.2.12.1 Notes sur le déroulement du test	134
	7.4.2.12.2 Conclusion du déroulement du test	134
	7.4.2.12.3 Validation du test	134
7.4.3	pyTv-S2-L2INIT-012-SENTINEL2-MUSCATE-CAMS_CHAIN	134
7.4.3.1	Titre	134
7.4.3.2	Objectif	134
7.4.3.3	Description	134
7.4.3.4	Liste des données d'entrées	135
7.4.3.5	Liste des produits de sortie	136
7.4.3.6	Prérequis	136
7.4.3.7	Durée attendue	136
7.4.3.8	Epsilon utilisé sur la non regression	136
7.4.3.9	Vérifications à effectuer	137
7.4.3.10	Mise en oeuvre du test	137
7.4.3.11	Exigences	137
7.4.3.12	Journal d'essai de la recette	137
	7.4.3.12.1 Notes sur le déroulement du test	137
	7.4.3.12.2 Conclusion du déroulement du test	137
	7.4.3.12.3 Validation du test	137
7.4.4	pyTv-S2-L2INIT-012-SENTINEL2-MUSCATE-CAMS_CHAIN	137
7.4.4.1	Titre	137
7.4.4.2	Objectif	137
7.4.4.3	Description	138
7.4.4.4	Liste des données d'entrées	138
7.4.4.5	Liste des produits de sortie	139

7.4.4.6	Prérequis	140
7.4.4.7	Durée attendue	140
7.4.4.8	Epsilon utilisé sur la non regression	140
7.4.4.9	Vérifications à effectuer	140
7.4.4.10	Mise en oeuvre du test	140
7.4.4.11	Exigences	140
7.4.4.12	Journal d'essai de la recette	140
7.4.4.12.1	Notes sur le déroulement du test	140
7.4.4.12.2	Conclusion du déroulement du test	140
7.4.4.12.3	Validation du test	140
7.4.5	pyTv-S2-L2NOMINAL-011-F-N-SENTINEL2_MUSCATE-GENERAL_CHAIN	141
7.4.5.1	Titre	141
7.4.5.2	Objectif	141
7.4.5.3	Description	141
7.4.5.4	Liste des données d'entrées	141
7.4.5.5	Liste des produits de sortie	142
7.4.5.6	Prérequis	142
7.4.5.7	Durée attendue	143
7.4.5.8	Epsilon utilisé sur la non regression	143
7.4.5.9	Vérifications à effectuer	143
7.4.5.10	Mise en oeuvre du test	143
7.4.5.11	Exigences	143
7.4.5.12	Journal d'essai de la recette	143
7.4.5.12.1	Notes sur le déroulement du test	143
7.4.5.12.2	Conclusion du déroulement du test	143
7.4.5.12.3	Validation du test	143
7.4.6	pyTv-S2-L2BACKWARD-011-F-N-SENTINEL2_MUSCATE-GENERAL_CHAIN	143
7.4.6.1	Titre	143
7.4.6.2	Objectif	144
7.4.6.3	Description	144
7.4.6.4	Liste des données d'entrées	144
7.4.6.5	Liste des produits de sortie	145
7.4.6.6	Prérequis	145
7.4.6.7	Durée attendue	145
7.4.6.8	Epsilon utilisé sur la non regression	145
7.4.6.9	Vérifications à effectuer	145
7.4.6.10	Mise en oeuvre du test	145
7.4.6.11	Exigences	145
7.4.6.12	Journal d'essai de la recette	146
7.4.6.12.1	Notes sur le déroulement du test	146
7.4.6.12.2	Conclusion du déroulement du test	146
7.4.6.12.3	Validation du test	146
7.4.7	pyTv-S2-L2NOMINAL-011-F-N-SENTINEL2_MUSCATE-MIXRES_CHAIN	146
7.4.7.1	Titre	146
7.4.7.2	Objectif	146
7.4.7.3	Description	147
7.4.7.4	Liste des données d'entrées	147
7.4.7.5	Liste des produits de sortie	148
7.4.7.6	Prérequis	148
7.4.7.7	Durée attendue	148
7.4.7.8	Epsilon utilisé sur la non regression	148
7.4.7.9	Vérifications à effectuer	148
7.4.7.10	Mise en oeuvre du test	149
7.4.7.11	Exigences	149

7.4.7.12	Journal d'essai de la recette	149
7.4.7.12.1	Notes sur le déroulement du test	149
7.4.7.12.2	Conclusion du déroulement du test	149
7.4.7.12.3	Validation du test	149
7.4.8	pyTv-S2-L2NOMINAL-011-F-N-SENTINEL2_MUSCATE-LUTDIRCOR_CHAIN	149
7.4.8.1	Titre	149
7.4.8.2	Objectif	149
7.4.8.3	Description	150
7.4.8.4	Liste des données d'entrées	150
7.4.8.5	Liste des produits de sortie	151
7.4.8.6	Prérequis	151
7.4.8.7	Durée attendue	151
7.4.8.8	Epsilon utilisé sur la non regression	151
7.4.8.9	Vérifications à effectuer	151
7.4.8.10	Mise en oeuvre du test	152
7.4.8.11	Exigences	152
7.4.8.12	Journal d'essai de la recette	152
7.4.8.12.1	Notes sur le déroulement du test	152
7.4.8.12.2	Conclusion du déroulement du test	152
7.4.8.12.3	Validation du test	152
7.4.9	pyTv-S2-L2INIT-CROSSWRITING-SENTINEL2-SENTINEL2MUSCATE	152
7.4.9.1	Titre	152
7.4.9.2	Objectif	152
7.4.9.3	Description	152
7.4.9.4	Liste des données d'entrées	153
7.4.9.5	Liste des produits de sortie	154
7.4.9.6	Prérequis	154
7.4.9.7	Durée attendue	154
7.4.9.8	Epsilon utilisé sur la non regression	154
7.4.9.9	Vérifications à effectuer	154
7.4.9.10	Mise en oeuvre du test	154
7.4.9.11	Exigences	154
7.4.9.12	Journal d'essai de la recette	154
7.4.9.12.1	Notes sur le déroulement du test	154
7.4.9.12.2	Conclusion du déroulement du test	154
7.4.9.12.3	Validation du test	155
7.4.10	pyTv-S2-L2INIT-CROSSWRITING-LANDSAT8-LANDSAT8MUSCATE	155
7.4.10.1	Titre	155
7.4.10.2	Objectif	155
7.4.10.3	Description	155
7.4.10.4	Liste des données d'entrées	155
7.4.10.5	Liste des produits de sortie	156
7.4.10.6	Prérequis	156
7.4.10.7	Durée attendue	156
7.4.10.8	Epsilon utilisé sur la non regression	156
7.4.10.9	Vérifications à effectuer	156
7.4.10.10	Mise en oeuvre du test	156
7.4.10.11	Exigences	156
7.4.10.12	Journal d'essai de la recette	157
7.4.10.12.1	Notes sur le déroulement du test	157
7.4.10.12.2	Conclusion du déroulement du test	157
7.4.10.12.3	Validation du test	157
7.4.11	pyTv-S2-L2INIT-CROSSWRITING-VENUS-VENUSMUSCATE	157
7.4.11.1	Titre	157

7.4.11.2	Objectif	157
7.4.11.3	Description	157
7.4.11.4	Liste des données d'entrées	157
7.4.11.5	Liste des produits de sortie	158
7.4.11.6	Prérequis	158
7.4.11.7	Durée attendue	158
7.4.11.8	Epsilon utilisé sur la non regression	158
7.4.11.9	Vérifications à effectuer	158
7.4.11.10	Mise en oeuvre du test	158
7.4.11.11	Exigences	159
7.4.11.12	Journal d'essai de la recette	159
7.4.11.12.1	Notes sur le déroulement du test	159
7.4.11.12.2	Conclusion du déroulement du test	159
7.4.11.12.3	Validation du test	159
7.4.12	pyTv-VE-L2INIT-VENUS-MUSCATE_CHAIN	159
7.4.12.1	Titre	159
7.4.12.2	Objectif	159
7.4.12.3	Description	159
7.4.12.4	Liste des données d'entrées	160
7.4.12.5	Liste des produits de sortie	160
7.4.12.6	Prérequis	160
7.4.12.7	Durée attendue	160
7.4.12.8	Epsilon utilisé sur la non regression	160
7.4.12.9	Vérifications à effectuer	161
7.4.12.10	Mise en oeuvre du test	161
7.4.12.11	Exigences	161
7.4.12.12	Journal d'essai de la recette	161
7.4.12.12.1	Notes sur le déroulement du test	161
7.4.12.12.2	Conclusion du déroulement du test	161
7.4.12.12.3	Validation du test	161
7.4.13	pyTv-STARTMAJA_CHAIN	161
7.4.13.1	Titre	161
7.4.13.2	Objectif	161
7.4.13.3	Description	162
7.4.13.4	Liste des données d'entrées	162
7.4.13.5	Liste des produits de sortie	162
7.4.13.6	Prérequis	162
7.4.13.7	Durée attendue	162
7.4.13.8	Epsilon utilisé sur la non regression	162
7.4.13.9	Vérifications à effectuer	163
7.4.13.10	Mise en oeuvre du test	163
7.4.13.11	Exigences	163
7.4.13.12	Journal d'essai de la recette	163
7.4.13.12.1	Notes sur le déroulement du test	163
7.4.13.12.2	Conclusion du déroulement du test	163
7.4.13.12.3	Validation du test	163

8 ANNEXES 165

Bibliography 167



CS Systèmes d'Information

Business Unit ESPACE

Département Payload Data & Applications

Image Processing

Software: MAJA			
SETG-PE-MAJA-020-CS			
Change	01	Date	30/04/2020
Issue	00	Date	30/04/2020
Distribution Code	E		
Ref. : CSSI/SPACE/PD&A/MAJA/PE			

Plan d'Intégration et de Validation [PE]

Rédigé par :	le : 30/04/2020
ESQUIS Benjamin CSSI/ESPACE/PDA/IP	
Pour application :	le : 30/04/2020
ESQUIS Benjamin CSSI/ESPACE/PDA/IP	

Bordereau d'indexation

Confidentialité : DLP	Mots clés : Conception, MAJA, Sentinel2, VENμS, chaînes L2		
TITRE Du Document : Plan d’Intégration et de Validation MAJA [PE]			
AUTEUR(s):	ESQUIS Benjamin CSSI/ESPACE/PDA/IP		
RESUME : Ce document présente l’ensemble des tests et moyens mis en œuvre par l’équipe de développement pour mener à bien les différents essais à mener au cours de la réalisation du projet MAJA			
Documents Rattaches : Le document vit seul	LOCALISATION : CSSI/SPACE/PD&A/MAJA		
Volume : 1	Nombre total de pages : N/A Dont pages préliminaires : 0 Nombre de pages suppl : 0	Doc composite : N	LANGUAGE : FR
GESTION DE CONF. : NG	CM RESP. :		
CAUSE D’EVOLUTION : Création pour MAJA v4.2.0			
CONTRAT : Marché ACSIS n°181112			

Diffusion interne

ESQUIS Benjamin	CSSI/ESPACE/PDA/IP
BROSSARD Julie	CSSI/ESPACE/PDA/IP
OLIVIE Francis	CSSI/ESPACE/DSM
RECULEAU SERGE	CSSI/ESPACE/PDA/PDGS

Diffusion externe

Name	Entity	Observations
BAILLARIN Simon	DNO/OT/IS	
HAGOLLE Olivier	DSO/SI/CB	
KETTIG Peter	DNO/OT/IS	
LARIF Marie-France	DNO/OT/PE	
SELLE Arnaud	DNO/OT/TA	
PACHOLCZYK Philippe	DNO/OT/TA	
PUJOL Mathilde	DNO/DA/AQ	

Modification

Ed.	Rév.	Date	Référence, Auteur(s), Causes d'évolution
01	00	30/04/2020	CSSI/ESPACE/PDA/IP/MAJA/PE ESQUIS Benjamin CSSI/ESPACE/PDA/IP Création pour MAJA 4.2

GLOSSAIRE ET LISTE DES PARAMÈTRES AC & AD

OT	Organigramme Technique
PDL	Program Design Language

Liste des paramètres AC :

Liste des paramètres AD :

GÉNÉRALITÉS

2.1 Documents de référence

L'ensemble de la documentation de référence au projet MAJA est décrite dans le document [LD] référencé SETG-LD-MAJA-010-CS correspondant à la liste documentaire du projet et présentant de façon exhaustive et précise l'ensemble de la documentation de référence du projet.

2.2 Documents applicables

L'ensemble de la documentation applicable au projet MAJA est décrite dans le document [LD] référencé SETG-LD-MAJA-010-CS correspondant à la liste documentaire du projet et présentant de façon exhaustive et précise l'ensemble de la documentation applicable au projet.

INTRODUCTION

3.1 objet du document

L'objectif du plan d'essai est de présenter l'ensemble des tests et moyens mis en œuvre par l'équipe de développement pour mener à bien les différents essais à mener au cours de la réalisation du projet MAJA.

3.2 structure du document

Ce document s'articule autour de quatre grandes parties :

- Présentation générale avec :
 - Ce même chapitre,
 - L'environnement de validation,
 - Les tests Unitaires : * Description des tests unitaires,
 - Les tests de validation algorithmique (Validation numérique) : * Description des tests, * Liste des tests, * Description détaillée des tests
 - Les essais de validation (Validation globale) : * Description de la phase d'Intégration et de Validation, * Liste des essais, * Description détaillée des essais
- Annexes

ENVIRONNEMENT DE VALIDATION

4.1 Configuration matérielle et dimensionnement

L'ensemble des essais se déroule à la fois sur la plateforme de développement CS, sur le cluster du CNES sur un noeuds 2019 en rh7 mais également sur les noeuds Cloud de l'intégration continue.

Il faut environ 180 Go d'espace disque pouvoir générer tous les tests des TV et des TVA, réparti comme ceci : - Pour les données d'entrées et les données de références (MAJA-Data) : 100 Go, - Pour les données produites après exécution de tous les TV : 20 Go de données de sortie et 60 Go de données de caching, soit 80 Go, - Pour les données produites après exécution de tous les TVA : 6 Go (incluant données de sortie et données de caching). Nous précisons de plus que lors des tests de régression entre les sorties de référence et celles générées par la chaîne, si les deux images comparées diffèrent, une image de différence est générée. Ceci provoque de facto une augmentation de la mémoire utilisée.

4.2 Moyens de validation

La phase de validation repose sur les moyens suivants : - Les plans d'essais contenant les scénarii prêts à être déroulés, - La configuration matérielle de développement, - La configuration matérielle cible, au CNES, équipée du socle technique, des logiciels de base et des COTS nécessaires à l'exploitation, afin de réaliser les essais de préintégration des chaînes de traitement L2 et L3, ainsi que les essais de performances, - Des outils de rejeu automatique des essais, permettant de vérifier les non-régressions, pour les traitements batch uniquement (cf outils ctest et CDash présenté section 3.5.2), - Des jeux de données (cf section suivante)

4.3 Outils

4.3.1 Moniteur de tests

L'environnement de test et de validation utilisé est « ctest » (outil intégré à CMake). En effet, avec CMake, il est extrêmement aisé de créer et d'ajouter des nouveaux tests dans un CMakeLists.txt (commande ADD_TEST), de les compiler et de les lancer. Cet environnement est notamment utilisé au quotidien par l'équipe de développement CS sur le projet Orfeo Toolbox, qui crée des tests automatiques. Cet outil permet la définition et l'enchaînement automatique (ou interactif) d'essais et ce, sur tous types de plates-formes et environnements de compilation. Cet outil assure un suivi de ces essais conformément aux usages dans ce type de développement en traçant les résultats sous forme de journal de bord. Cet outil permet d'automatiser un grand nombre de tâches de validation fastidieuses, et facilite le rejeu d'essais de non-régression pendant les phases de développement, de validation, de garantie et de maintenance. Cet outil est donc utilisé pour lancer les tests unitaires et de validation.

4.3.2 Exploitation des résultats des tests : Serveur CDash et SonarCube

L'outil CDash permet d'exploiter les résultats produits par l'outil de test ctest. L'outil SonarCube permet de mesurer en continue la qualité du code.

4.4 Données de Validation

4.4.1 Venus

Une série de produits sur le site CAMARGUE de niveau 1 (avec et sans nuage) , décalés de deux jours, est disponible pour les essais : - Produit image 3 : VE_TEST_VSC_L1VALD_CAMARGUE_20120101 - Produit image 4 : VE_TEST_VSC_L1VALD_CAMARGUE_20120103 Les produits précédents contiennent des valeurs de réflectance très élevées et non réalistes. Les pixels sont donc tous déclarés nuageux par l'algorithme de détection des nuages présents dans la chaîne L2. Pour pouvoir poursuivre les traitements après le module de détection des nuages, il est nécessaire d'augmenter le seuil sur la couverture nuageuse figurant dans les GIPP (GIP_L2COMM). Les tests réalisés sur ces données sont donc des tests d'ordre fonctionnel et ne permettent pas de valider numériquement les algorithmes. Des données météo/ozone sont utilisées pour corriger les images de réflectance de l'absorption gazeuse. - Donnée ozone du type VE_TEST_EXO_METDTA_20110101T000000_20111231T230000.DBL La donnée MNT utilisée est la suivante : - Donnée MNT du type VE_TEST_AUX_REFDE2_CAMARGUE_0001.DBL Cette donnée a été corrigée pour la version 4.2 de MACCS est respecte désormais la convention "centre de pixel".

4.4.2 Cas Venus (au format "muscate")

Ces images Venus permettent de valider la chaîne MAJA avec des produits issus de ce capteur mais au format MUSCATE. Ces images sont au format défini dans l'interface du PSC-SL-411-0032 (contenant des fichiers tif pour les images de réflectances, des masques et des fichiers xml).

4.4.3 Sentinel 2 (S2A et S2B)

Les images SENTINEL2 peuvent également être traitées dans la chaîne MAJA. Ces images sont à des résolutions différentes en fonction des bandes spectrales et permettent donc de valider dans la chaîne : - La lecture spécifique d'un produit SENTINEL2 et la conversion au format image des polygones décrits dans des fichiers gml, - la prise en compte de résolutions multiples dans un même produit, - les algorithmes de détection des cirrus (génération d'un masque de cirrus spécifique aux produits SENTINEL2) et de la neige, - la prise en compte d'angles de prise de vue variables en fonction des détecteurs lors de la correction de la diffusion.

Les produits de niveau 1 contiennent : - un fichier JPEG2000 par bande spectrale - S2A_TEST_IMG_GPPL1C_065_20091211165851_20091211165932_0 - un fichier descriptif du produit au format xml - S2A_TEST_MTD_GPPL1C_065_20091211165851_20091211165932_0 2.4. Données de validation 11 - tout un ensemble de fichiers gml contenant notamment les masques des détecteurs, des détecteurs aberrants, des pixels saturés et des no_data sous forme de polygones :

- S2A_TEST_MSK_DEFECT_GPPL1C_065_20091211165851_20091211165932_00_000000_15SVD_0003.gml
- S2A_TEST_MSK_DETFOO_GPPL1C_065_20091211165851_20091211165932_00_000000_15SVD_0003.gml
- S2A_TEST_MSK_NODATA_GPPL1C_065_20091211165851_20091211165932_00_000000_15SVD_0003.gml
- S2A_TEST_MSK_SATURA_GPPL1C_065_20091211165851_20091211165932_00_000000_15SVD_0003.gml
- S2A_TEST_MSK_TECQUA_GPPL1C_065_20091211165851_20091211165932_00_000000_15SVD_0003.gml

4.4.4 Cas Sentinel2 (au format “muscate”)

Ces images Sentinel2 permettent de valider la chaîne MAJA avec des produits issus de ce capteur mais au format MUSCATE. Ces images sont au format défini dans l'interface du PSC-SL-411-0032 (contenant des fichiers tif pour les images de réflectances, des masques et des fichiers xml).

4.4.5 Cas Landsat 8 (au format MUSCATE)

Ces images LANDSAT L8 permettent de valider la chaîne MAJA avec des produits issus de ce capteur mais au format MUSCATE. Ces images sont au format défini dans l'interface du PSC-SL-411-0032 (contenant des fichiers tif pour les images de réflectances, des masques et des fichiers xml).

4.4.6 Cas Landsat 8 (au format “natif”)

Les produits de niveau 1 utilisés sont les suivants. Attention : Pour un même “site”, les produits à des dates différentes sont superposables géographiquement mais n'ont pas la même emprise. Ces conditions ne respectent pas les conditions imposés par MAJA à savoir que les données d'entrées (L1, L2 et MNT) misent en entrée de MAJA doivent être superposables entre elles (géographiquement) avec rigoureusement la même emprise (tailles d'images en nombres de pixels). Les données utilisées en entrées sont les suivantes : - LC81980302013193LGN00 - LC81980302013177LGN01 Elles ont été générées par le projet S2-Agri afin qu'elles respectent les conditions de MAJA. Les MNTs associés (ASP, MSK, SLP et ALT) ont été générés par le projet S2-Agri.

4.5 Installation de l'environnement de validation

4.5.1 Pré-requis

Pour l'environnement de validation il est nécessaire d'avoir une version de développement de MAJA disponible sur le dépôt du code et de suivre les instructions de configuration et de compilation décrites dans le document “USER, INSTALLATION, and OPERATING MANUAL, MAJA L2 & L3CHAINS” (fichier “SETG-MU-MAJA-010-CS”) en activant les tests à l'aide des variables `ENABLE_TV`, `ENABLE_TVA` et `ENABLE_TU`. Il est également nécessaire de récupérer l'ensemble des données de validation disponibles sur le cluster du CNES.

4.5.2 Lancement des tests

Une fois maja compilé l'ensemble des tests est disponible en tapant la commande “ctest -N”. Un test particulier peut être lancé à l'aide de la commande “ctest -R <nomdutest>”

LES TESTS UNITAIRES

5.1 Objectifs

L'objectif des tests unitaires est de :

- Réaliser une validation fonctionnelle des traitements codés,
- Réaliser de façon naturelle et ascendante une intégration des couches basses du logiciel qui sont disponibles.

Chaque répertoire (composant logiciel) contient l'ensemble des tests unitaires mis en place permettant de valider les classes développées du composant logiciel concerné.

5.2 Stratégie et méthode des tests unitaires

Ces tests unitaires ne sont pas intégrés au [PE]. Ils seront toutefois mis à jour et maintenus suite aux évolutions introduites dans les phases suivantes, pour faciliter notamment la phase de maintenance.

L'outil « ctest » permet d'activer les traitements conformément à la stratégie définie ci-dessus. Les contextes de jeux de données sont conservés avec le module lanceur, de même que les résultats d'exécution qui sont mis en référence une fois le TU validé. Les résultats des tests, soumis sur le serveur cdash, sont par conséquent archivés et consultables aisément sous la forme du dashboard (cf. [DJ] pour une présentation du dashboard).

A la fin de la phase de développement, plus de 700 tests unitaires ont été mis en place. Ces tests unitaires couvrent l'ensemble des composants logiciels développés et assurent une intégration complète de MAJA. Pour chaque classe développée, au minimum deux tests unitaires sont mis en place et ce dès le début du codage de la classe. Pour la validation numérique des algorithmes (ex : interpolation de LUTs, algorithme de Levenberg-Marquartz, etc.), CS utilise autant que faire se peut, des données fournies par le CNES et référencées dans la base gérée en configuration. Chaque algorithme peut ainsi être validé en utilisant en entrée les produits intermédiaires (simple format TIFF) issus de la maquette développée par le CESBIO.

L'outil de comparaison de l'OTB effectue une différence des images pixel à pixel. Pour chaque bande, il génère l'image différence, calcule la somme des différences. Le test est validé si ces valeurs sont nulles à un epsilon près, qui est un paramètre d'entrée du test. L'outil calcule également le nombre de pixels sur lesquels sont mesurées ces différences. Il peut être intéressant de calculer si cet écart est inférieur à epsilon pour chaque pixel de manière à ne pas masquer une différence importante mais locale sur l'image. Une analyse visuelle des images et des images de différence est réalisée. Elle permet, par exemple, de détecter une structuration non aléatoire des différences sur l'image. Les différences obtenues entre les images sont alors analysées pour en connaître l'origine et déterminer s'il s'agit d'une différence de méthode (lié par exemple à l'utilisation d'interpolateurs ou d'outils de corrélation différents) ou d'un problème lié au développement. Il convient également de déterminer si la différence mesurée est acceptable par rapport à la donnée testée (ordre de grandeur des écarts par rapport aux valeurs issues du prototype).

L'objectif des tests unitaires est de couvrir tous les cas possibles :

- cas nominaux,
- cas aux limites.

Une fois l'algorithme validé, les résultats du test sont référencés et dès lors, systématiquement contrôlés à chaque nouvelle exécution du test. Cette démarche assure la non-régression des algorithmes développés tout au long de la phase de codage. Les résultats des tests joués sont systématiquement envoyés au serveur CDash pour être mis en forme et offrir au développeur un environnement de contrôle ergonomique et permettre le suivi de l'évolution des résultats au cours du temps.

5.3 Formalisation des tests unitaires

Les tests unitaires ne font pas l'objet d'une description textuelle formelle au travers d'une fiche ou d'un plan d'essai.

LES TESTS DE VALIDATION ALGORITHMIQUE

6.1 Objectifs

L'objectif des tests de validation algorithmique est de : - Réaliser une validation numérique des traitements algorithmiques de "haut niveau" codés sous forme d'applications, - Vérifier le comportement de ces traitements dans certains cas dégradés ou aux limites qui ne seront pas repris en validation, avec le responsable algorithmique CNES. Chaque module algorithmique suivant fait l'objet d'un ou plusieurs tests permettant de couvrir l'ensemble des fonctionnalités de chaque composant :

6.2 Stratégie et méthode des tests numériques

Les données Formosat sont principalement utilisées pour la validation numérique (à l'exception du test de détection des cirrus qui nécessite le plan d'altitude stéréoscopique des nuages présent dans les produits Venus). Suite au développement de la chaîne MAJA, des tests supplémentaires avec des produits Landsat et Sentinel2 ont été ajoutés pour prendre en compte les dernières évolutions de la chaîne comme la création de nouveaux algorithmes (détection de la neige, des cirrus, estimation des aérosols, etc.) ou la gestion de produits à multiples résolutions. L'essentiel de la validation numérique est menée en comparant chaque image du produit de niveau 2 générée par les chaînes scientifiques Venus à celle produite précédemment par le prototype développé au CESBIO. L'outil de comparaison de l'OTB effectue une différence des images pixel à pixel. Pour chaque bande, il génère l'image des différences et calcule la somme des différences. Le test est validé si ces valeurs sont nulles à un epsilon près, qui est un paramètre d'entrée du test. L'outil calcule également le nombre de pixels sur lesquels sont mesurés ces différences. Compte tenu de la complexité des algorithmes, la valeur nulle semble difficile à atteindre. La valeur du epsilon sera fixée en concertation avec le CNES et en tenant compte des résultats obtenus pour les tests unitaires. Il convient également de déterminer si la différence mesurée est acceptable par rapport à la donnée testée (ordre de grandeur des écarts par rapport aux valeurs issues du prototype). Comme pour les tests unitaires, une analyse importante est l'analyse visuelle des images et des images de différence. Elle permet de détecter une structuration non aléatoire des différences sur l'image mais aussi de localiser les différences (proche d'un nuage, d'une zone d'eau, etc.). A cette analyse des images de différence, s'ajoutent d'autres moyens d'analyse. Une stratégie de validation différente est adoptée en fonction du types de données à comparer. Plusieurs types de données sont comparés : les images contenant des grandeurs physiques, les images de dates et les masques. La validation des produits composites passe dans un premier temps par la comparaison des plans « date » pour visualiser les zones qui diffèrent. Les différences mesurées lors de la comparaison des images de date vont se retrouver sur les images composite de réflectance. La validation des masques présents dans les produits L2 se traduit par une analyse statistique des différences entre le masque généré et le masque de référence si besoin est. En effet, la probabilité d'avoir des masques parfaitement identiques entre les masques issus des chaînes Venus et les masques de référence est faible. Pour comparer ces masques, la simple différence réalisée pixel à pixel peut être complétée par la mise en place d'indicateurs statistiques. Ces derniers permettent de quantifier la performance des détections. Les performances du masque obtenu sont alors exprimées par une matrice de confusion par rapport au masque de référence. A ces indicateurs s'ajoutent le calcul des proportions de bonnes et de fausses détections c'est-à-dire le nombre de pixels correctement identifiés par rapport au nombre de pixels de l'image. Une fois l'algorithme validé, les résultats

du test sont référencés et dès lors, systématiquement contrôlés à chaque nouvelle exécution du test. Cette démarche assure la non-régression des algorithmes développés tout au long de la phase de codage. Les résultats des tests joués sont systématiquement envoyés au serveur CDash pour être mis en forme et offrir au développeur un environnement de contrôle ergonomique et le suivi de l'évolution des résultats au cours du temps.

6.3 Formalisation des tests de validation algorithmique

La nomenclature adoptée pour l'identification des tests algorithmiques est la suivante : apTv<traitement><nature> Ex : apTvAOTEstimation_SpectroTemporalNominal

6.4 LISTE DES TESTS DE VALIDATION ALGORITHMIQUE

Ce chapitre présente la liste des tests mis en place pour réaliser la validation numérique des modules algorithmiques.

1	apTvSmoothingBis	Application d'une gaussienne sur l'image
2	apTvResampling	Rééchantillonnage d'une image sur un DTM
3	apTvStats	Calculs de statistiques sur une image
4	apTvMultiplyByScalar	Multiplication d'une image par un scalaire
5	apTvBinaryConcatenate	Concaténation de plusieurs bandes binaires(0-1)
6	apTvBinaryToVector	Déconcaténation en plan de bits
7	apTvConcatenatePerZone	Concaténation d'image selon un masque de zones
8	apTvDispatchZonesToVectorNoWeight	Répartition de masques de zone
9	apTvDispatchZonesToVectorWeight	Répartition de masques de zone (bandes non ordonnées)
10	apTvIncidenceAngle	Calcul des angles d'incidences
11	apTvOzoneExtractor	Extraction du volume d'ozone
12	apTvAngleListToImage	Création d'une image à partir d'une liste d'angles
13	apTvSolarAngle	Calcul des angles solaires
14	apTvImageListWriter	Ecriture simultanée d'une liste d'images
15	apTvRoundImage	Arrondis d'une image
16	apTvRoundVectorImage	Arrondis d'une image à plusieurs bandes
17	apTvRescaleIntensity	Modification de la radiometrie
18	apTvOneBandEqualThreshold	Création d'un masque par test de valeur
19	apTvDarkestPixelAOTNominal	Calcul du DarkestPixel pour le calcul de l'AOT
20	apTvGapFilling	Algorithme de gap filling
21	apTvAOTEstimation_SpectroTemporalNominal	Calcul de l'AOT en SpectroTemporel
22	apTvAOTMean_SpectroTemporalNominal	Calcul de l'AOT moyen en SpectroTemporel
23	apTvAOTEstimation_MultiTemporalNominal	Calcul de l'AOT en MultiTemporel
24	apTvAOTEstimationMultiSpectral	Calcul de l'AOT en MultiSpectral
25	apTvAOTEstimationSpectroTemporalCAMS	Calcul de l'AOT en SpectroTemporel avec CAMS
26	apTvPressureFilter	Calcul de la pression
27	apTvTOACorrection	Correction TOA de l'image
28	apTvWaterAmountGeneratorFilter	Calcul d'un volume d'eau
29	apTvGammaCompute	Calcul d'un gamma pour la correction cirrus
30	apTvCirrusCorrectionApply	Correction Cirrus d'une image
31	apTvCirrusFlag	Calcul du flag Cirrus
32	apTvCirrusMask	Calcul du masque de Cirrus
33	apTvComposite	Calcul des images composites
34	apTvDTMProcessing	Algorithme DTMProcessing

continues on next page

Table 1 – continued from previous page

35	apTvComputeRhoEnv	Calcul du RhoEnv
36	apTvEnvCorrection	Correction des effets environnementaux
37	apTvInterpolateForEnvCorr	Interpolation spécifique pour les effets environnementaux
38	apTvCloudReflectance	Calcul du masque de nuages reflectance
39	apTvCloudReflectanceVariation	Calcul du masque de variation des nuages reflectance
40	apTvCloudExtension	Calcul du masque de nuages "extension"
41	apTvCloudShadVar	Calcul du masque de variation des ombres des nuages
42	apTvCloudSnowCorrection	Correction des effets de neiges
43	apTvCloudMaskUndersampling	Reechantillonnage spécifique aux images CLA
44	apTvCloudShadowWithAltitudeFilterInit	Calcul du masque des ombres des nuages connaissant leurs altitudes
45	apTvCloudShadowWithoutAltitudeFilterNominal	Calcul du masque des ombres des nuages sans leurs altitudes
46	apTvRayleighCorrection	Correction Rayleigh
47	apTvCamsCompute	Extraction des données CAMS (ancien format)
48	apTvNewCamsCompute	Extraction des données CAMS (nouveau format)
49	apTvNewCamsMixedCompute	Extraction des données CAMS (ancien et nouveau format)
50	apTvDirectionalCorrectionROY	Calcul des coefficients de correction directionnelle (modèle ROY)
51	apTvDirectionalCorrectionLUT	Calcul des coefficients de correction directionnelle (modèle LUT)
52	apTvReduceLut	Extraction de MiniLut
53	apTvReduceLutDirCorr	Extraction de minilut avec correction directionnelle
54	apTvExtractLutChannel	Extraction d'une bande d'une LUT
55	apTvScatteringCorrection	Correction du scattering
56	apTvSlopeCorrection	Correction des effets de pentes
57	apTvComputeTransmission	Calcul de la transmission atmosphérique
58	apTvSnowMask	Calcul du masque de neige
59	apTvSnowMaskVenus	Calcul du masque de neige (cas VENUS)
60	apTvWaterMask	Calcul du masque d'eau
61	apTvWaterVaporPostPro	Post processing du masque d'eau
62	apTvBandMath	Outil générique d'algorithmie sur image (OTB)
63	apTvExtractROI	Extraction de zone/bande (OTB)
64	apTvManageNoData	Application/Extraction de masques de NoData
65	apTvBinaryMorphologicalOperation	Opérateurs Morphologiques
66	apTvConvertCartoToGeoPoint	Conversion de coordonnées cartographiques
67	apTvConcatenateImages	Concaténation d'image en une seule

6.5 DESCRIPTION DÉTAILLÉE DES TESTS DE VALIDATION ALGORITHMIQUE

6.5.1 apTvAngleListToImage

6.5.1.1 Objectif

Validation de l'application "AngleListToImage"

6.5.1.2 Description

Le module “AngleListToImage” calcul une image à partir d’une liste d’angles. Cela permet notamment de créer une image des angles à partir de métadonnées (S2).

6.5.1.3 Liste des données d’entrées

- Les angles sous format texte selon la nomenclature d’interface
- Le DTM sur lequel les angles doivent être calculés

6.5.1.4 Liste des produits de sortie

Image contenant les angle d’incidence pour chaque pixel

6.5.1.5 Prérequis

Il n’y a pas de prérequis.

6.5.1.6 Durée attendue

La durée d’exécution de l’essai n’est pas un critère attendu.

6.5.1.7 Epsilon utilisé sur la non regression

0

6.5.1.8 Vérifications à effectuer

Le test génère en sortie une image contenant les angles.

6.5.1.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : `cctest -R apTvAngleListToImage`

6.5.1.10 Journal d’essai de la recette

6.5.1.10.1 Notes sur le déroulement du test

Rien de particulier n’a été noté lors du déroulement du test.

6.5.1.10.2 Conclusion du déroulement du test

RAS

6.5.1.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.1.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.2 apTvAOTEstimationMultiSpectral

6.5.2.1 Objectif

Validation de l'application "AOTEstimation" en mode init Multi-Spectral

6.5.2.2 Description

Le test "AOTEstimationMultiSpectral" calcul la valeur de l'AOT en mode init Multi-Spectral. Le module AOTEstimation génère une image contenant pour chaque pixel, l'épaisseur optique des aérosols estimée.

6.5.2.3 Liste des données d'entrées

- toac AOTEstimation/MultiSpectral/L930662_20020325_L7_197_030_USGS_toa_240m.tif
- shd AOTEstimation/MultiSpectral/L930662_20020325_L7_197_030_USGS_toa_240m.shd
- dtm DTM/LANDSAT/VE_TEST_AUX_REFDE2_L930662__0001.DBL.DIR/L930662_240m.mnt_ALC
- cld AOTEstimation/MultiSpectral/L930662_20020325_L7_197_030_USGS_toa_240m.nua
- edg AOTEstimation/MultiSpectral/L930662_20020325_L7_197_030_USGS_toa_bord_240m
- sat AOTEstimation/MultiSpectral/L930662_20020325_L7_197_030_USGS_toa_240m.sat
- was AOTEstimation/MultiSpectral/L930662_20020325_L7_197_030_USGS_toa_240m.water
- snow AOTEstimation/MultiSpectral/L930662_20020325_L7_197_030_USGS_toa_240m.nei
- lutheader AOTEstimation/MultiSpectral/giplut.xml
- lutimage AOTEstimation/MultiSpectral/lutAOTMS.mha
- mode multispectral
- xmlparamsfile AOTEstimation/MultiSpectral/AOTParams_LANDSAT.xml
- reall2nodata 0.
- darkestaot 0.13
- jday 6617

- init true

6.5.2.4 Liste des produits de sortie

- aotmask TVA/apTvAOTEstimationMultiSpectral_aotmask.tif
- aotcomputed TVA/apTvAOTEstimationMultiSpectral_aotcomputed.tif

6.5.2.5 Prérequis

Il n'y a pas de prérequis.

6.5.2.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.2.7 Epsilon utilisé sur la non regression

0

6.5.2.8 Vérifications à effectuer

Le test génère en sortie l'image des valeurs de l'AOT ainsi que le masque des valeurs calculées ou non.

6.5.2.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvAOTEstimationMultiSpectral

6.5.2.10 Journal d'essai de la recette

6.5.2.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.2.10.2 Conclusion du déroulement du test

RAS

6.5.2.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.2.11 Exigences

Ce test couvre les exigences suivantes : ST-169-AOT-0010 (C) ; ST-169-AOT-0100 (C) ; ST-169-AOT-0200 (C) ; ST-169-AOT-0300 (C) ; ST-169-AOT-0500 (C) ; ST-169-AOT-0600 (C) ;

6.5.3 apTvAOTEstimation_MultiTemporalNominal

6.5.3.1 Objectif

Validation de l'application "AOTEstimation" en mode nominal Multi-Temporel

6.5.3.2 Description

Le test "AOTEstimation_MultiTemporalNominal" calcul la valeur de l'AOT en mode nominal Multi-Temporel. Le module AOTEstimation génère une image contenant pour chaque pixel, l'épaisseur optique des aérosols estimée.

6.5.3.3 Liste des données d'entrées

- toac AOTEstimation/MultiTemporal/Sudouest_20061114_MS_fmsat_ortho_toac.tif
- shd AOTEstimation/MultiTemporal/Sudouest_20061114_MS_fmsat_ortho_shd.tif
- dtm AOTEstimation/MultiTemporal/Sudouest_20061114_MS_fmsat_ortho_dtm.tif
- cld AOTEstimation/MultiTemporal/Sudouest_20061114_MS_fmsat_ortho_cld.tif
- l2ndt AOTEstimation/MultiTemporal/Sudouest_20061114_MS_fmsat_ortho_IL2inNDT.tif
- l2rtc AOTEstimation/MultiTemporal/Sudouest_20061114_MS_fmsat_ortho_IL2inRTC.tif
- edg AOTEstimation/MultiTemporal/Sudouest_20061114_MS_fmsat_ortho_edg.tif
- sat AOTEstimation/MultiTemporal/Sudouest_20061114_MS_fmsat_ortho_sat.tif
- l2rta AOTEstimation/MultiTemporal/Sudouest_20061114_MS_fmsat_ortho_IL2inRTA.tif
- l2pxd AOTEstimation/MultiTemporal/Sudouest_20061114_MS_fmsat_ortho_IL2inPXD.tif
- was AOTEstimation/MultiTemporal/Sudouest_20061114_MS_fmsat_ortho_was.tif
- snow AOTEstimation/MultiTemporal/Sudouest_20061114_MS_fmsat_ortho_snw.tif
- lutheader AOTEstimation/MultiTemporal/giplut.xml
- lutimage AOTEstimation/MultiTemporal/lutAOTMT1.mha
- mode multitemporal
- xmlparamsfile AOTEstimation/MultiTemporal/AOTParams_LANDSAT.xml
- temporal.altitudemean 0
- jday 8036
- reall2nodata 0
- darkestat 0.0430164
- temporal.lutmap AOTEstimation/MultiTemporal/giplutmap.xml

6.5.3.4 Liste des produits de sortie

- aotmask TVA/apTvAOTEstimationMultitemporalNominal_aotmask.tif
- aotcomputed TVA/apTvAOTEstimationtemporalNominal_aotcomputed.tif

6.5.3.5 Prérequis

Il n'y a pas de prérequis.

6.5.3.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.3.7 Epsilon utilisé sur la non regression

0

6.5.3.8 Vérifications à effectuer

Le test génère en sortie l'image des valeurs de l'AOT ainsi que le masque des valeurs calculées ou non.

6.5.3.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvAOTEstimation_MultiTemporalNominal

6.5.3.10 Journal d'essai de la recette

6.5.3.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.3.10.2 Conclusion du déroulement du test

RAS

6.5.3.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.3.11 Exigences

Ce test couvre les exigences suivantes : ST-169-AOT-0010 (C) ; ST-169-AOT-0100 (C) ; ST-169-AOT-0200 (C) ; ST-169-AOT-0300 (C) ; ST-169-AOT-0500 (C) ; ST-169-AOT-0700 (C) ; ST-169-AOT-0800 (C) ; ST-169-AOT-0600 (C) ; ST-169-AOToversampling-0010 (C) ;

6.5.4 apTvAOTEstimationMultiTemporalCAMS

6.5.4.1 Objectif

Validation de l'application "AOTEstimation" en mode nominal Spectro-Temporel en utilisant des données CAMS

6.5.4.2 Description

Le test "AOTEstimationMultiTemporalCAMS" calcul la valeur de l'AOT en mode nominal Spectro-Temporel en utilisant les données CAMS. Le module AOTEstimation génère une image contenant pour chaque pixel, l'épaisseur optique des aérosols estimée.

6.5.4.3 Liste des données d'entrées

- toac AOTEstimation/SpectroTemporalCAMS/normal/20160614/PRIVE/toa_abs/31TFJR108_20160614_S2A_L1Csimu_toa_240m.tif
- shd AOTEstimation/SpectroTemporalCAMS/normal/20160614/PRIVE/mnt/31TFJR108_20160614_S2A_L1Csimu_toa_240m.tif
- dtm DTM/SENTINEL2/S2__OPER_AUX_REFDE2_31TFJ____0001.DBL.DIR/S2__OPER_AUX_REFDE2_31TFJ____0001.tif
- cld AOTEstimation/SpectroTemporalCAMS/normal/20160614/PRIVE/nua/31TFJR108_20160614_S2A_L1Csimu_toa_240m.tif
- l2ndt AOTEstimation/SpectroTemporalCAMS/MasqueNoir.tif
- l2rtc AOTEstimation/SpectroTemporalCAMS/normal/20160607/PRIVE/surf_comp/31TFJR008_20160607_S2A_L1Csimu_surf.tif
- edg AOTEstimation/SpectroTemporalCAMS/normal/20160614/PRIVE/bord/31TFJR108_20160614_S2A_L1Csimu_toa_bord_240m.tif
- lutheader AOTEstimation/SpectroTemporalCAMS/giplut.xml
- lutimage AOTEstimation/SpectroTemporalCAMS/lutAOT.mha
- mode spectrotemporal
- xmlparamsfile AOTEstimation/SpectroTemporalCAMS/AOTParams_SENTINEL2.xml
- temporal.altitudemean 182.243020728
- reall2nodata -10
- jday 11853
- darkestaot 0.123605
- temporal.lutmap AOTEstimation/SpectroTemporalCAMS/giplutmap.xml
- sat AOTEstimation/SpectroTemporalCAMS/MasqueNoir.tif
- l2rta AOTEstimation/SpectroTemporalCAMS/normal/20160607/PRIVE/toa_comp/31TFJR008_20160607_S2A_L1Csimu_toa_comp.tif
- l2pxd AOTEstimation/SpectroTemporalCAMS/normal/20160607/PRIVE/date/31TFJR008_20160607_S2A_L1Csimu_toa_240m.tif
- was AOTEstimation/SpectroTemporalCAMS/normal/20160614/PRIVE/water/31TFJR108_20160614_S2A_L1Csimu_toa_240m.tif
- snow AOTEstimation/SpectroTemporalCAMS/normal/20160614/PRIVE/nei/31TFJR108_20160614_S2A_L1Csimu_toa_240m.tif

- cams.aot AOTEstimation/SpectroTemporalCAMS/ReduceLutComputationCAMSTest001-AOT.tif
- cams.altmean 182.243020728
- cams.nodatatau -1

6.5.4.4 Liste des produits de sortie

- aotmask TVA/apTvAOTEstimationSpectrotemporalCams_aotmask.tif
- aotcomputed TVA/apTvAOTEstimationSpectrotemporalCams_aotcomputed.tif
- aotcams TVA/apTvAOTEstimationSpectrotemporalCams_aotcams.tif

6.5.4.5 Prérequis

Il n'y a pas de prérequis.

6.5.4.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.4.7 Epsilon utilisé sur la non regression

0

6.5.4.8 Vérifications à effectuer

Le test génère en sortie l'image des valeurs de l'AOT ainsi que le masque des valeurs calculées ou non.

6.5.4.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvAOTEstimationMultiTemporalCAMS

6.5.4.10 Journal d'essai de la recette

6.5.4.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.4.10.2 Conclusion du déroulement du test

RAS

6.5.4.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.4.11 Exigences

Ce test couvre les exigences suivantes : ST-169-AOT-0010 (C) ; ST-169-AOT-0100 (C) ; ST-169-AOT-0200 (C) ; ST-169-AOT-0300 (C) ; ST-169-AOT-0500 (C) ; ST-169-AOT-0700 (C) ; ST-169-AOT-0800 (C) ; ST-169-AOT-0600 (C) ; ST-169-AOToversampling-0010 (C) ;

6.5.5 apTvAOTEstimation_SpectroTemporalNominal

6.5.5.1 Objectif

Validation de l'application "AOTEstimation" en mode nominal Spectro-Temporel

6.5.5.2 Description

Le test "AOTEstimation_SpectroTemporalNominal" calcul la valeur de l'AOT en mode nominal Spectro-Temporel. Le module AOTEstimation génère une image contenant pour chaque pixel, l'épaisseur optique des aérosols estimée.

6.5.5.3 Liste des données d'entrées

- toac AOTEstimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_toac_240m.tif
- shd AOTEstimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_shd_240m.tif
- dtm DTM/LANDSAT/LN_TEST_AUX_REFDE2_EU93066200A00B_0001.DBL.DIR/LN_TEST_AUX_REFDE2_EU93066200A00B_0001.DBL
- cld AOTEstimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_l2cld_240m.tif
- l2ndt AOTEstimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_l2ndt_240m.tif
- l2rtc AOTEstimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_l2rtc_240m.tif
- edg AOTEstimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_edg_240m.tif
- lutheader AOTEstimation/SpectroTemporal/giplut.xml
- lutimage AOTEstimation/SpectroTemporal/lutAOT.mha
- mode spectrotemporal
- xmlparamsfile AOTEstimation/SpectroTemporal/AOTParams_LANDSAT.xml
- temporal.altitudemean 759.93489999999970532371662557
- reall2nodata -10
- jday 6617
- darkestat 0.11594
- temporal.lutmap AOTEstimation/SpectroTemporal/giplutmap.xml
- sat AOTEstimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_sat_240m.tif

- l2rta AOTEstimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_l2rta_240m.tif
- l2pxd AOTEstimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_l2pxd_240m.tif
- was AOTEstimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_was_240m.tif
- snow AOTEstimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_snow_240m.tif

6.5.5.4 Liste des produits de sortie

- aotmask TVA/apTvAOTEstimationSpectrotemporalNominal_aotmask.tif
- aotcomputed TVA/apTvAOTEstimationSpectrotemporalNominal_aotcomputed.tif

6.5.5.5 Prérequis

Il n'y a pas de prérequis.

6.5.5.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.5.7 Epsilon utilisé sur la non regression

0

6.5.5.8 Vérifications à effectuer

Le test génère en sortie l'image des valeurs de l'AOT ainsi que le masque des valeurs calculées ou non.

6.5.5.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvAOTEstimation_SpectroTemporalNominal

6.5.5.10 Journal d'essai de la recette

6.5.5.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.5.10.2 Conclusion du déroulement du test

RAS

6.5.5.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.5.11 Exigences

Ce test couvre les exigences suivantes : ST-169-AOT-0010 (C) ; ST-169-AOT-0100 (C) ; ST-169-AOT-0200 (C) ; ST-169-AOT-0300 (C) ; ST-169-AOT-0500 (C) ; ST-169-AOT-0600 (C) ;

6.5.6 apTvAOTMean_SpectroTemporalNominal

6.5.6.1 Objectif

Validation de l'application "AOTEstimation" en mode nominal Spectro-Temporel

6.5.6.2 Description

Le test "AOTMean_SpectroTemporalNominal" calcul la valeur de l'AOT moyen en mode nominal Spectro-Temporel.

6.5.6.3 Liste des données d'entrées

- toac AOTEstimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_toac_240m.tif
- shd AOTEstimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_shd_240m.tif
- dtm DTM/LANDSAT/LN_TEST_AUX_REFDE2_EU93066200A00B_0001.DBL.DIR/LN_TEST_AUX_REFDE2_EU93066200A00B_0001.DBL
- cld AOTEstimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_l2cld_240m.tif
- l2ndt AOTEstimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_l2ndt_240m.tif
- l2rtc AOTEstimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_l2rtc_240m.tif
- edg AOTEstimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_edg_240m.tif
- lutheader AOTEstimation/SpectroTemporal/giplut.xml
- lutimage AOTEstimation/SpectroTemporal/lutAOT.mha
- mode spectrotemporal
- xmlparamsfile AOTEstimation/SpectroTemporal/AOTParams_LANDSAT.xml
- temporal.altitudemean 759.93489999999970532371662557
- reall2nodata -10
- jday 6617
- darkestaot 0.11594
- temporal.lutmap AOTEstimation/SpectroTemporal/giplutmap.xml
- sat AOTEstimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_sat_240m.tif
- l2rta AOTEstimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_l2rta_240m.tif

- l2pxd AOTestimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_l2pxd_240m.tif
- was AOTestimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_was_240m.tif
- snow AOTestimation/SpectroTemporal/L930662_20020212_L7_198_030_USGS_snow_240m.tif

6.5.6.4 Liste des produits de sortie

- Valeurs de l'AOT moyen

6.5.6.5 Prérequis

Il n'y a pas de prérequis.

6.5.6.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.6.7 Epsilon utilisé sur la non regression

0

6.5.6.8 Vérifications à effectuer

Le test calcul la valeurs moyenne de l'AOT sur l'image.

6.5.6.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvAOTMean_SpectroTemporalNominal

6.5.6.10 Journal d'essai de la recette

6.5.6.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.6.10.2 Conclusion du déroulement du test

RAS

6.5.6.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.6.11 Exigences

Ce test couvre les exigences suivantes : ST-169-AOT-0010 (C) ; ST-169-AOT-0100 (C) ; ST-169-AOT-0200 (C) ; ST-169-AOT-0300 (C) ; ST-169-AOT-0500 (C) ; ST-169-AOT-0600 (C) ;

6.5.7 apTvBinaryConcatenate

6.5.7.1 Objectif

Validation de l'application "BinaryConcatenate"

6.5.7.2 Description

Le module "BinaryConcatenate" concatene les bandes d'un masque à plusieurs bandes en une seule bande dont chaque bit représente une bande d'entrée.

6.5.7.3 Liste des données d'entrées

Image de type masque ayant plusieurs bandes

6.5.7.4 Liste des produits de sortie

Image de type masque ayant une seule bande dont chaque bit représente une bande d'entrée

6.5.7.5 Prérequis

Il n'y a pas de prérequis.

6.5.7.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.7.7 Epsilon utilisé sur la non regression

0

6.5.7.8 Vérifications à effectuer

Le test génère en sortie une image ayant une seule bande dont chaque bit représente une bande d'entrée .

6.5.7.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvBinaryConcatenate

6.5.7.10 Journal d'essai de la recette

6.5.7.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.7.10.2 Conclusion du déroulement du test

RAS

6.5.7.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.7.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.8 apTvBinaryToVector

6.5.8.1 Objectif

Validation de l'application "BinaryToVector"

6.5.8.2 Description

Le module "BinaryToVector" split les différents bit d'une image masque multi-bit en autant d'image monobit contenant chacune un bit de l'image d'entrée.

6.5.8.3 Liste des données d'entrées

Image de type masque multibit sur un octet ayant une seule bande

6.5.8.4 Liste des produits de sortie

Image de type masque ayant une plusieurs bande dont chacune représente un bit de masque d'entrée

6.5.8.5 Prérequis

Il n'y a pas de prérequis.

6.5.8.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.8.7 Epsilon utilisé sur la non regression

0

6.5.8.8 Vérifications à effectuer

Le test génère en sortie une image ayant une plusieurs bande dont chacune représente un bit de masque d'entrée.

6.5.8.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvBinaryToVector

6.5.8.10 Journal d'essai de la recette

6.5.8.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.8.10.2 Conclusion du déroulement du test

RAS

6.5.8.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.8.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.9 apTvCamsCompute

6.5.9.1 Objectif

Validation de l'application "CamsCompute"

6.5.9.2 Description

Le module "CamsCompute" extrait les valeurs de pondération cams d'un fichier CAMS.

6.5.9.3 Liste des données d'entrées

- Fichier CAMS
- Lat/lon souhaité

6.5.9.4 Liste des produits de sortie

Valeurs de pondération des modèles pour la localisation demandée

6.5.9.5 Prérequis

Il n'y a pas de prérequis.

6.5.9.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.9.7 Epsilon utilisé sur la non regression

0

6.5.9.8 Vérifications à effectuer

Les valeurs de pondération doivent correspondre avec la zone

6.5.9.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvCamsCompute

6.5.9.10 Journal d'essai de la recette

6.5.9.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.9.10.2 Conclusion du déroulement du test

RAS

6.5.9.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.9.11 Exigences

Ce test couvre les exigences suivantes : ST-169-ReducedLUT-0010 (C) ;

6.5.10 apTvCirrusCorrectionApply

6.5.10.1 Objectif

Validation de l'application "CirrusCorrection"

6.5.10.2 Description

Le test "CirrusCorrection" applique la correction Cirrus à une image.

6.5.10.3 Liste des données d'entrées

- l2toa CirrusCorrectionTest/11SLT_20150803_S2A_L1Csimu_toa_abs_coeff_001_10m.tif
- l2edg CirrusCorrectionTest/11SLT_20150803_S2A_L1Csimu_bord_10m.bord_final
- l2dtm DTM/SENTINEL2/11SLT/11SLT_10m_roi.tif
- l2cirrus CirrusCorrectionTest/11SLT_20150803_S2A_L1Csimu_toa_60m.B10_resampled_coef_001.tif
- cirrusmask apTv11SLT_20150803_mask_cirrus_10m.tif
- resolution 10
- maxdistance 1000
- thresholdfactor 0.95
- thresholdoffset 0.015
- thresholdgain 0.00002
- gammamean 1.055496216
- swirgammafactor 2
- maxreflectance 0.07
- bandcodelist B2 B3 B4 B8
- swirbandcodelist B11 B12

6.5.10.4 Liste des produits de sortie

- correctedtoa TVA/apTv11SLT_20150803_corrected_TOA_10m.tif

6.5.10.5 Prérequis

Il n'y a pas de prérequis.

6.5.10.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.10.7 Epsilon utilisé sur la non regression

0

6.5.10.8 Vérifications à effectuer

Le test génère en sortie l'image corrigée.

6.5.10.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvCirrusCorrectionApply

6.5.10.10 Journal d'essai de la recette

6.5.10.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.10.10.2 Conclusion du déroulement du test

RAS

6.5.10.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.10.11 Exigences

Ce test couvre les exigences suivantes : ST-169-CirrusCorr-0010 (C) ;

6.5.11 apTvCirrusFlag

6.5.11.1 Objectif

Validation de l'application "CirrusFlag"

6.5.11.2 Description

Le module "CirrusFlag" calcul le flag cirrus du produit.

6.5.11.3 Liste des données d'entrées

- cla CirrusFlagTest/CAMARGUE_20120103_CLA_sub.tif
- edg CirrusFlagTest/CAMARGUE_20120103_VE_ortho_toa_bord_100m
- altthreshold 1000
- minpercentcirrus 10
- minpercentcloud 15
- minstdcirrus 10
- minpercentstdcirrus 500
- nodata -10

6.5.11.4 Liste des produits de sortie

- flag cirrus

6.5.11.5 Prérequis

Il n'y a pas de prérequis.

6.5.11.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.11.7 Epsilon utilisé sur la non regression

0

6.5.11.8 Vérifications à effectuer

Le test génère en sortie le flag cirrus du produit.

6.5.11.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvCirrusFlag

6.5.11.10 Journal d'essai de la recette

6.5.11.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.11.10.2 Conclusion du déroulement du test

RAS

6.5.11.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.11.11 Exigences

Ce test couvre les exigences suivantes : ST-169-CirrusFlagVE-0010 (C) ; ST-169-CirrusFlagVE-0020 (C) ; ST-169-CirrusFlagVE-0030(C) ; ST-169-CirrusFlagVE-0040 (C) ;

6.5.12 apTvCirrusMask

6.5.12.1 Objectif

Validation de l'application "CirrusMask"

6.5.12.2 Description

Le module "CirrusMask" calcul le masque cirrus du produit.

6.5.12.3 Liste des données d'entrées

- toa CirrusMaskTest/15SVD_20091211_S2A_L1Csimu_toa_240m_coef_001.tif
- edg CirrusMaskTest/15SVD_20091211_S2A_L1Csimu_toa_bord_240m
- dtm DTM/SENTINEL2/VE_TEST_AUX_REFDE2_15SDV____0001.DBL.DIR/S2N_TEST_AUX_REFDE2_15SVD_0001_AI
- bandidx 10
- threshoff 0.35
- threshgain 0
- minpercent 10
- altref 2000

6.5.12.4 Liste des produits de sortie

- mask TVA/apTv15SVD_20091211_S2A_L1Csimu_CirrusMask.tif

6.5.12.5 Prérequis

Il n'y a pas de prérequis.

6.5.12.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.12.7 Epsilon utilisé sur la non regression

0

6.5.12.8 Vérifications à effectuer

Le test génère en sortie le masque cirrus du produit.

6.5.12.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvCirrusMask

6.5.12.10 Journal d'essai de la recette

6.5.12.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.12.10.2 Conclusion du déroulement du test

RAS

6.5.12.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.12.11 Exigences

Ce test couvre les exigences suivantes : ST-169-CirrusFlagS2-0020 (C) ; ST-169-CirrusFlagS2-0030 (C) ;

6.5.13 apTvCloudReflectance

6.5.13.1 Objectif

Validation de l'application "CloudExtension"

6.5.13.2 Description

Le module "CloudExtension" calcul le masque d'extension des nuages.

6.5.13.3 Liste des données d'entrées

- tocr CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_tocr.tif
- edg CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_edg.tif
- cldext apTvReflectanceVariation_ReflExt.tif
- cldreftl apTvReflectanceVariation_Refl.tif
- cldreftlvar apTvReflectanceVariation_ReflVar.tif
- l2rcr CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_l2rcr.tif
- l2sto CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_l2sto.tif
- l2ndt CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_l2ndt.tif
- nodata -10
- correlbandtocr 0
- correlbandrcr 0
- jday 8260
- minpixelcorrelation 15
- correlthreshold 0.900000000000000022204460492503
- stolistofdates 0 0 0 0 0 0 0 0 0
- ncorrel 3 3
- discardedrationmin 0.2

6.5.13.4 Liste des produits de sortie

- ext TVA/apTvCloudExtension.tif

6.5.13.5 Prérequis

Il n'y a pas de prérequis.

6.5.13.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.13.7 Epsilon utilisé sur la non regression

0

6.5.13.8 Vérifications à effectuer

Le test génère en sortie le masque extension du produit.

6.5.13.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvCloudExtension

6.5.13.10 Journal d'essai de la recette

6.5.13.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.13.10.2 Conclusion du déroulement du test

RAS

6.5.13.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.13.11 Exigences

Ce test couvre les exigences suivantes : ST-169-CloudMasking-0300 (C) ;ST-169-CloudMasking-0010 (C) ; ST-169-CloudMasking-0100 (C) ; ST-169-CloudMasking-0200(C) ; ST-169-CloudMasking-0500 (C) ; ST-169-CloudMasking-0600 (C) ; ST-169-CloudMasking-0700 (C) ; ST-169-CloudMasking-1000 (C) ; ST-169-CloudMasking-1600 (C) ; ST-169-CloudMasking-1700 (C) ; ST-169-CloudMasking-0300 (C) ; ST-169-CloudMasking-0400 (C) ;ST-169-CloudMasking-0410 (C) ; ST-169-CloudMasking-0420 (C) ; ST-169-CloudMasking-0430(C) ; ST-169-CloudMasking-1100 (C) ; ST-169-CloudMasking-1110 (C) ; ST-169-CloudMasking-1120 (C) ; ST-169-CloudMasking-1130 (C) ; ST-169-CloudMasking-1140 (C) ; ST-169-CloudMasking-1150 (C) ; ST-169-CloudMasking-1160 (C) ; ST-169-CloudMasking-1170 (C) ;ST-169-CloudMasking-1180 (C) ; ST-169-CloudMasking-1200 (C) ; ST-169-CloudMasking-1300(C) ; ST-169-CloudMasking-1400 (C) ; ST-169-CloudMasking-1500 (C) ; ST-169-CloudMasking-1510 (C) ; ST-169-CloudMasking-1520 (C) ;

6.5.14 apTvCloudMaskUndersampling

6.5.14.1 Objectif

Validation de l'application "CloudMaskUndersampling"

6.5.14.2 Description

Le module "CloudMaskUndersampling" rééchantillonne les image CLA sur un DTM donné.

6.5.14.3 Liste des données d'entrées

- Image CLA
- DTM

6.5.14.4 Liste des produits de sortie

Image lambda rééchantillonnée sur l'emprise du DTM avec l'interpolateur spécial CLA

6.5.14.5 Prérequis

Il n'y a pas de prérequis.

6.5.14.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.14.7 Epsilon utilisé sur la non regression

0

6.5.14.8 Vérifications à effectuer

Le test génère en sortie une image dont l'emprise est calée sur le DTM

6.5.14.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvCloudMaskUndersampling

6.5.14.10 Journal d'essai de la recette

6.5.14.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.14.10.2 Conclusion du déroulement du test

RAS

6.5.14.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.14.11 Exigences

Ce test couvre les exigences suivantes : ST-169-CloudMasking-0010 (C) ; ST-169-CloudMasking-0100 (C) ; ST-169-CloudMasking-0200) ; ST-169-CloudMasking-0500 (C) ; ST-169-CloudMasking-0600 (C) ; ST-169-CloudMasking- (C) ; ST-169-CloudMasking-1000 (C) ; ST-169-CloudMasking-1600 (C) ; ST-169-CloudMasking-1700 (C) ; ST-169-CloudMasking-0300 (C) ; ST-169-CloudMasking-0400 (C) ;ST-169-CloudMasking-0410 (C) ; ST-169-CloudMasking-0420 (C) ; ST-169-CloudMasking-0430(C) ; ST-169-CloudMasking-1100 (C) ; ST-169-CloudMasking-1110 (C) ; ST-169-CloudMasking-CS - Communication & Systèmes — SETG-PE-MAJA-010-1120 (C) ; ST-169-CloudMasking-1130 (C) ; ST-169-CloudMasking-1140 (C) ; ST-169-CloudMasking-1150 (C) ; ST-169-CloudMasking-1160 (C) ; ST-169-CloudMasking-1170 (C) ;ST-169-CloudMasking-1180 (C) ; ST-169-CloudMasking-1200 (C) ; ST-169-CloudMasking-1300(C) ; ST-169-CloudMasking-1400 (C) ; ST-169-CloudMasking-1500 (C) ; ST-169-CloudMasking-1510 (C) ; ST-169-CloudMasking-1520 (C) ;

6.5.15 apTvCloudReflectance

6.5.15.1 Objectif

Validation de l'application "CirrusReflectance"

6.5.15.2 Description

Le module "CirrusReflectance" calcul le masque de reflectance de nuages.

6.5.15.3 Liste des données d'entrées

- tocr CloudMaskingTest/Sudouest_20060326_MS_fmsat_ortho_tocr.tif
- edg CloudMaskingTest/Sudouest_20060326_MS_fmsat_ortho_edg.tif
- sat CloudMaskingTest/Sudouest_20060326_MS_fmsat_ortho_sat.tif
- initmode true
- foregroundvalue 1
- bluebandtocr 0
- redbandtocr 2
- nirbandtocr 3
- reflectancewaterband 0
- redreflectancethresholdvalue 0.14999999999999994448884876874
- bluereflectancethresholdvalue 0.25
- nirredrationmaxthresholdvalue 2
- nirredrationminthresholdvalue 0.800000000000000044408920985006
- nodata -10
- correlbandtocr 0
- waterthreshold 1
- swir.bandtocr 0
- swir.nirswirrationmin 1

6.5.15.4 Liste des produits de sortie

- refl TVA/apTv31TCJ_20150904_S2A_L1Csimu_cloud_reflectance.tif

6.5.15.5 Prérequis

Il n'y a pas de prérequis.

6.5.15.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.15.7 Epsilon utilisé sur la non regression

0

6.5.15.8 Vérifications à effectuer

Le test génère en sortie le masque reflectance du produit.

6.5.15.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvCloudReflectance

6.5.15.10 Journal d'essai de la recette

6.5.15.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.15.10.2 Conclusion du déroulement du test

RAS

6.5.15.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.15.11 Exigences

Ce test couvre les exigences suivantes : ST-169-CloudMasking-0300 (C) ;ST-169-CloudMasking-0010 (C) ; ST-169-CloudMasking-0100 (C) ; ST-169-CloudMasking-0200(C) ; ST-169-CloudMasking-0500 (C) ; ST-169-CloudMasking-0600 (C) ; ST-169-CloudMasking-0700 (C) ; ST-169-CloudMasking-1000 (C) ; ST-169-CloudMasking-1600 (C) ; ST-169-CloudMasking-1700 (C) ; ST-169-CloudMasking-0300 (C) ;

ST-169-CloudMasking-0400 (C) ;ST-169-CloudMasking-0410 (C) ; ST-169-CloudMasking-0420 (C) ; ST-169-CloudMasking-0430(C) ; ST-169-CloudMasking-1100 (C) ; ST-169-CloudMasking-1110 (C) ; ST-169-CloudMasking-1120 (C) ; ST-169-CloudMasking-1130 (C) ; ST-169-CloudMasking-1140 (C) ; ST-169-CloudMasking-1150 (C) ; ST-169-CloudMasking-1160 (C) ; ST-169-CloudMasking-1170 (C) ;ST-169-CloudMasking-1180 (C) ; ST-169-CloudMasking-1200 (C) ; ST-169-CloudMasking-1300(C) ; ST-169-CloudMasking-1400 (C) ; ST-169-CloudMasking-1500 (C) ; ST-169-CloudMasking-1510 (C) ; ST-169-CloudMasking-1520 (C) ;

6.5.16 apTvCloudReflectance

6.5.16.1 Objectif

Validation de l'application "CirrusReflectanceVariation"

6.5.16.2 Description

Le module "CirrusReflectanceVariation" calcul le masque de variation reflectance de nuages.

6.5.16.3 Liste des données d'entrées

- tocr CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_tocr.tif
- sat CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_sat.tif
- edg CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_edg.tif
- l2was CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_l2was.tif
- l2rcr CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_l2rcr.tif
- l2sto CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_l2sto.tif
- l2pxd CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_l2pxd.tif
- l2ndt CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_l2ndt.tif
- initmode false
- foregroundvalue 1
- bluebandtocr 0
- redbandtocr 2
- nirbandtocr 3
- reflectancewaterband 2
- blureflectancethresholdvalue 0.200000000000000011102230246252
- redreflectancethresholdvalue 0.14999999999999994448884876874
- nirredratiomaxthresholdvalue 2
- nirredratiominthresholdvalue 0.800000000000000044408920985006
- nodata -10
- correlbandtocr 0
- watrthreshold 0.0800000000000000016653345369377

- nominal.seradius 1
- nominal.bluebandrcr 0
- nominal.redbandrcr 2
- nominal.nirbandrcr 3
- nominal.correlbandrcr 0
- nominal.jday 8260
- nominal.cloudforgettingduration 30
- nominal.minvarthreshold 0.029999999999999988897769753748
- nominal.maxvarthreshold 0.089999999999999966693309261245
- nominal.knirblue 1
- nominal.kredblue 1.5
- nominal.deltathreshold 0.00500000000000000010408340855861
- nominal.minpixelcorrelation 15
- nominal.correlthreshold 0.900000000000000022204460492503
- nominal.ncorrel 3 3
- nominal.stolistofdates 0 0 0 0 0 0 0 0 0
- swir.bandtocr 0
- swir.nirswirrationmin 1

6.5.16.4 Liste des produits de sortie

- refl TVA/apTvReflectanceVariation_Refl.tif
- reflvar TVA/apTvReflectanceVariation_ReflVar.tif
- reflxt TVA/apTvReflectanceVariation_ReflExt.tif

6.5.16.5 Prérequis

Il n'y a pas de prérequis.

6.5.16.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.16.7 Epsilon utilisé sur la non regression

0

6.5.16.8 Vérifications à effectuer

Le test génère en sortie le masque reflectance variation du produit.

6.5.16.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvCloudReflectanceVariation

6.5.16.10 Journal d'essai de la recette**6.5.16.10.1 Notes sur le déroulement du test**

Rien de particulier n'a été noté lors du déroulement du test.

6.5.16.10.2 Conclusion du déroulement du test

RAS

6.5.16.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.16.11 Exigences

Ce test couvre les exigences suivantes :

ST-169-CloudMasking-0010 (C) ; ST-169-CloudMasking-0100 (C) ; ST-169-CloudMasking-0200(C) ; ST-169-CloudMasking-0500 (C) ; ST-169-CloudMasking-0600 (C) ; ST-169-CloudMasking-0700 (C) ; ST-169-CloudMasking-1000 (C) ; ST-169-CloudMasking-1600 (C) ; ST-169-CloudMasking-1700 (C) ; ST-169-CloudMasking-0300 (C) ; ST-169-CloudMasking-0400 (C) ;ST-169-CloudMasking-0410 (C) ; ST-169-CloudMasking-0420 (C) ; ST-169-CloudMasking-0430(C) ; ST-169-CloudMasking-1100 (C) ; ST-169-CloudMasking-1110 (C) ; ST-169-CloudMasking-1120 (C) ; ST-169-CloudMasking-1130 (C) ; ST-169-CloudMasking-1140 (C) ; ST-169-CloudMasking-1150 (C) ; ST-169-CloudMasking-1160 (C) ; ST-169-CloudMasking-1170 (C) ;ST-169-CloudMasking-1180 (C) ; ST-169-CloudMasking-1200 (C) ; ST-169-CloudMasking-1300(C) ; ST-169-CloudMasking-1400 (C) ; ST-169-CloudMasking-1500 (C) ; ST-169-CloudMasking-1510 (C) ;ST-169-CloudMasking-1520 (C) ;

6.5.17 apTvCloudShadowWithAltitudeFilterInit

6.5.17.1 Objectif

Validation de l'application "CloudShadowWithAltitudeFilterInit"

6.5.17.2 Description

Le module "CloudShadowWithAltitudeFilterInit" calcul le masque d'ombre des nuages.

6.5.17.3 Liste des données d'entrées

- tocr CloudMaskingTest/Sudouest_20060326_MS_fmsat_ortho_tocr.tif
- edg CloudMaskingTest/Sudouest_20060326_MS_fmsat_ortho_edg.tif
- cldall CloudMaskingTest/Sudouest_20060326_MS_fmsat_ortho_cldall.tif
- cla CloudMaskingTest/Sudouest_20060326_MS_fmsat_ortho_cla.tif
- vie CloudMaskingTest/Sudouest_20060326_MS_fmsat_ortho_vie.tif
- dtm DTM/FORMOSAT/VE_TEST_AUX_REFDE2_SUDOUEST_0001.DBL.DIR/Sudouest_96m.mnt_ALC
- initmode true
- sol1.in CloudMaskingTest/Sudouest_20060326_MS_fmsat_ortho_sol1.tif
- sol1.h 2000
- solhref 1000
- defaultalt 500
- ksigma 3
- l2coarseres 96
- deltahmax 8000
- deltahmin 0
- deltahstep 500
- shadbandtocr 0
- shadbandrcr 0
- viehref 3000
- nodata -10
- algo.withalt.refinement 1
- algo.withalt.absnbpixthresh 20
- algo.withalt.threshdiffim 10
- algo.withalt.sol2.h 0
- algo.withoutalt.cloudcovermult 2
- algo.withoutalt.shadratiomax 0
- algo.withoutalt.shadratiomin 0

- algo.withoutalt.shadratiopixnb 0
- algo.withoutalt.maxdarkening 0
- algo.withoutalt.minpercentile 0
- algo.withoutalt.seradius 0

6.5.17.4 Liste des produits de sortie

- shadow TVA/apTvSudouest_20060326_MS_fmsat_ortho_cloudShadowWithAltitude_init_shadow.tif
- l2cla TVA/apTvSudouest_20060326_MS_fmsat_ortho_cloudShadowWithAltitude_init_l2cla.tif

6.5.17.5 Prérequis

Il n'y a pas de prérequis.

6.5.17.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.17.7 Epsilon utilisé sur la non regression

0

6.5.17.8 Vérifications à effectuer

Le test génère en sortie le masque d'ombre des nuages.

6.5.17.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvCloudShadowWithAltitudeFilterInit

6.5.17.10 Journal d'essai de la recette

6.5.17.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.17.10.2 Conclusion du déroulement du test

RAS

6.5.17.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.17.11 Exigences

Ce test couvre les exigences suivantes : ST-169-CloudMasking-0010 (C) ; ST-169-CloudMasking-0100 (C) ; ST-169-CloudMasking-0200(C) ; ST-169-CloudMasking-0500 (C) ; ST-169-CloudMasking-0600 (C) ; ST-169-CloudMasking-0700 (C) ; ST-169-CloudMasking-1000 (C) ; ST-169-CloudMasking-1600 (C) ; ST-169-CloudMasking-1700 (C) ; ST-169-CloudMasking-0300 (C) ; ST-169-CloudMasking-0400 (C) ;ST-169-CloudMasking-0410 (C) ; ST-169-CloudMasking-0420 (C) ; ST-169-CloudMasking-0430(C) ; ST-169-CloudMasking-1100 (C) ; ST-169-CloudMasking-1110 (C) ; ST-169-CloudMasking-1120 (C) ; ST-169-CloudMasking-1130 (C) ; ST-169-CloudMasking-1140 (C) ; ST-169-CloudMasking-1150 (C) ; ST-169-CloudMasking-1160 (C) ; ST-169-CloudMasking-1170 (C) ;ST-169-CloudMasking-1180 (C) ; ST-169-CloudMasking-1200 (C) ; ST-169-CloudMasking-1300(C) ; ST-169-CloudMasking-1400 (C) ; ST-169-CloudMasking-1500 (C) ; ST-169-CloudMasking-1510 (C) ; ST-169-CloudMasking-1520 (C) ;

6.5.18 apTvCloudShadowWithoutAltitudeFilterNominal

6.5.18.1 Objectif

Validation de l'application "CloudShadowWithoutAltitudeFilterNominal"

6.5.18.2 Description

Le module "CloudShadowWithoutAltitudeFilterNominal" calcul le masque d'ombre des nuages.

6.5.18.3 Liste des données d'entrées

- tocr CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_tocr.tif
- edg CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_edg.tif
- cldall CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_cldall.tif
- cla CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_cla.tif
- vie CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_vie.tif
- dtm DTM/FORMOSAT/Lacrau_96m.mnt
- initmode 0
- l2rcr CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_rcr.tif
- l2was CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_was.tif
- l2ndt CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_ndt.tif
- sol1.in CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_sol1.tif
- sol1.h 3000
- solhref 4000

- defaultalt 500
- ksigma 0
- l2coarseres 96
- deltahmax 8000
- deltahmin 0
- deltahstep 250
- shadbandtoer 2
- shadbandrer 2
- viehref 3000
- nodata -10
- algo withoutalt
- algo.withoutalt.cloudcovermult 2
- algo.withoutalt.shadratiomax 1
- algo.withoutalt.shadratiomin 0.5
- algo.withoutalt.shadratiopixnb 10
- algo.withoutalt.maxdarkening 0.800000000000000044408920985006
- algo.withoutalt.minpercentile 0.100000000000000005551115123126
- algo.withoutalt.seradius 2

6.5.18.4 Liste des produits de sortie

- shadow TVA/apTvLacrau_20060813_MS_fmsat_ortho_cloudShadowWithoutAltitude_nominal_shadow.tif

6.5.18.5 Prérequis

Il n'y a pas de prérequis.

6.5.18.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.18.7 Epsilon utilisé sur la non regression

0

6.5.18.8 Vérifications à effectuer

Le test génère en sortie le masque d'ombre des nuages.

6.5.18.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvCloudShadowWithoutAltitudeFilterNominal

6.5.18.10 Journal d'essai de la recette

6.5.18.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.18.10.2 Conclusion du déroulement du test

RAS

6.5.18.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.18.11 Exigences

Ce test couvre les exigences suivantes : ST-169-CloudMasking-0010 (C) ; ST-169-CloudMasking-0100 (C) ; ST-169-CloudMasking-0200(C) ; ST-169-CloudMasking-0500 (C) ; ST-169-CloudMasking-0600 (C) ; ST-169-CloudMasking-0700 (C) ; ST-169-CloudMasking-1000 (C) ; ST-169-CloudMasking-1600 (C) ; ST-169-CloudMasking-1700 (C) ; ST-169-CloudMasking-0300 (C) ; ST-169-CloudMasking-0400 (C) ;ST-169-CloudMasking-0410 (C) ; ST-169-CloudMasking-0420 (C) ; ST-169-CloudMasking-0430(C) ; ST-169-CloudMasking-1100 (C) ; ST-169-CloudMasking-1110 (C) ; ST-169-CloudMasking-1120 (C) ; ST-169-CloudMasking-1130 (C) ; ST-169-CloudMasking-1140 (C) ; ST-169-CloudMasking-1150 (C) ; ST-169-CloudMasking-1160 (C) ; ST-169-CloudMasking-1170 (C) ;ST-169-CloudMasking-1180 (C) ; ST-169-CloudMasking-1200 (C) ; ST-169-CloudMasking-1300(C) ; ST-169-CloudMasking-1400 (C) ; ST-169-CloudMasking-1500 (C) ; ST-169-CloudMasking-1510 (C) ; ST-169-CloudMasking-1520 (C) ;

6.5.19 apTvCloudShadVar

6.5.19.1 Objectif

Validation de l'application "CirrusShadVar"

6.5.19.2 Description

Le module "CirrusShadVar" calcul les variations sur le masque d'ombre des nuages.

6.5.19.3 Liste des données d'entrées

- tocr CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_tocr.tif
- edg CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_edg.tif
- cldshad CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_cldshad.tif
- cldall CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_cldall.tif
- vie CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_vie.tif
- dtm DTM/FORMOSAT/Lacrau_96m.mnt
- sol CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_sol.tif
- l2rcr CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_l2rcr.tif
- l2was CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_l2was.tif
- l2ndt CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_l2ndt.tif
- l2shad CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_shad.tif
- l2shadvar CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_shad_var.tif
- l2sto CloudMaskingTest/Lacrau_20060813_MS_fmsat_ortho_l2sto.tif
- l2coarseres 96
- shadbandtocr 2
- shadbandrcr 2
- viehref 3000
- solhref 3000
- nodata -10
- maxdarkening 0.8
- minpercentile 0.1
- maxcla 8226
- compositelabelratio 0.800000000000000044408920985006
- minpixelcorrelation 15
- correlthreshold 0.900000000000000022204460492503
- ncorrel 3 3
- stolistofdates 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
- jday 8260

6.5.19.4 Liste des produits de sortie

- shadvar TVA/apTvCloudShadVar.tif

6.5.19.5 Prérequis

Il n'y a pas de prérequis.

6.5.19.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.19.7 Epsilon utilisé sur la non regression

0

6.5.19.8 Vérifications à effectuer

Le test génère en sortie le masque de variation des ombres de nuages du produit.

6.5.19.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvCloudShadVar

6.5.19.10 Journal d'essai de la recette

6.5.19.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.19.10.2 Conclusion du déroulement du test

RAS

6.5.19.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.19.11 Exigences

Ce test couvre les exigences suivantes :

6.5.20 apTvCloudSnowCorrection

6.5.20.1 Objectif

Validation de l'application "CloudSnowCorrection"

6.5.20.2 Description

Le module "CloudSnowCorrection" calcul le masque de correction des effets de neige en prenant en compte les nuages.

Le module "SnowMaskCorrection" permet de corriger les fausses détection de neige située à l'intérieur des nuages. Les pixels neigeux entièrement entourés de nuages sont déclarés nuageux plutôt que neigeux.

6.5.20.3 Liste des données d'entrées

- snowin CloudMaskingTest/L930863_20130815_L8_196_030_USGS_toa_240m.neige_avant_correction
- edg CloudMaskingTest/L930863_20130815_L8_196_030_USGS_toa_bord_240m
- cldreflvar CloudMaskingTest/L930863_20130815_L8_196_030_USGS_toa_240m.refl_avant_correction
- cldrefl CloudMaskingTest/L930863_20130815_L8_196_030_USGS_toa_240m.reflvar_avant_correction
- seradius 1

6.5.20.4 Liste des produits de sortie

- snow TVA/apTvCloudSnowCorrection_snow.tif
- refl TVA/apTvCloudSnowCorrection_reflvar.tif

6.5.20.5 Prérequis

Il n'y a pas de prérequis.

6.5.20.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.20.7 Epsilon utilisé sur la non regression

0

6.5.20.8 Vérifications à effectuer

Le test génère en sortie le masque de correction des effets de neige.

6.5.20.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvCloudSnowCorrection

6.5.20.10 Journal d'essai de la recette

6.5.20.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.20.10.2 Conclusion du déroulement du test

RAS

6.5.20.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.20.11 Exigences

Ce test couvre les exigences suivantes : ST-169-CloudMasking-0400(C);ST-169-CloudMasking-0500(C);

6.5.21 apTvComposite

6.5.21.1 Objectif

Validation de l'application "Composite"

6.5.21.2 Description

Le module "Composite" calcul les images composites d'un produit. Le module "CompositeImage" est chargé de la mise à jour des images composites contenues dans la partie privée des produits de niveau 2

6.5.21.3 Liste des données d'entrées

- tocr CompositeTest/apTvSudouest_20060326_tocr.tif
- toac CompositeTest/apTvSudouest_20060326_toa.tif
- toc CompositeTest/apTvSudouest_20060326_toc.tif
- edg CompositeTest/apTvSudouest_20060326_edg.tif
- cldsum CompositeTest/apTvSudouest_20060326_pcld.tif
- sat CompositeTest/apTvSudouest_20060326_pdat.tif
- aot CompositeTest/apTvSudouest_20060326_aot_mask.tif
- initmode true
- reall2nodata -10
- l2nodata -10000
- sigmaaot 1
- minaot 0.10000000000000000555111512312
- maxaot 0.5
- maxage 60
- hotspotflag 0
- cirrusflag 0
- rainflag 0
- jday 8120
- stobandnumber 10
- stobandindex 0
- aotradius 1

6.5.21.4 Liste des produits de sortie

- rta TVA/apTvCompositeImageFilter_RTA.tif
- rcr TVA/apTvCompositeImageFilter_RCR.tif
- rtc TVA/apTvCompositeImageFilter_RTC.tif
- sto TVA/apTvCompositeImageFilter_STO.tif
- pxd TVA/apTvCompositeImageFilter_PXD.tif
- ndt TVA/apTvCompositeImageFilter_NDT.tif

6.5.21.5 Prérequis

Il n'y a pas de prérequis.

6.5.21.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.21.7 Epsilon utilisé sur la non regression

0

6.5.21.8 Vérifications à effectuer

Le test génère en sortie les images composites

6.5.21.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvComposite

6.5.21.10 Journal d'essai de la recette

6.5.21.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.21.10.2 Conclusion du déroulement du test

RAS

6.5.21.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.21.11 Exigences

Ce test couvre les exigences suivantes : ST-169-Composite-0010 (C) ; ST-169-Composite-0020 (C) ; ST-169-Composite-0030 (C) ; ST-169- Composite-0040 (C) ; ST-169-Composite-0050 (C) ;

6.5.22 apTvComputeRhoEnv

6.5.22.1 Objectif

Validation de l'application "ComputeRhoEnv"

6.5.22.2 Description

Le module "RhoEnv" calcul le Rho pour la correction environnementale.

6.5.22.3 Liste des données d'entrées

- toc EnvCorrection/Sudouest_20060329_MS_fmsat_ortho_surf_96m_tocSub.tif
- edg EnvCorrection/Sudouest_20060329_MS_fmsat_ortho_surf_96m_edgSub.tif
- radius 10
- filtercoeff 1

6.5.22.4 Liste des produits de sortie

- rhoenv TVA/apTvComputeRhoEnv.tif

6.5.22.5 Prérequis

Il n'y a pas de prérequis.

6.5.22.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.22.7 Epsilon utilisé sur la non regression

0

6.5.22.8 Vérifications à effectuer

Le test génère en sortie les images composites

6.5.22.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvComputeRhoEnv

6.5.22.10 Journal d'essai de la recette

6.5.22.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.22.10.2 Conclusion du déroulement du test

RAS

6.5.22.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.22.11 Exigences

Ce test couvre les exigences suivantes :

6.5.23 apTvComputeTransmission

6.5.23.1 Objectif

Validation de l'application "ComputeTransmission"

6.5.23.2 Description

Le test "ComputeTransmission" calcul les images de transmission pour la correction des effets de pentes à une image.

6.5.23.3 Liste des données d'entrées

- luttdir SlopeCorrection/giplutTdirmap.xml
- luttdif SlopeCorrection/giplutTdifmap.xml
- aot SlopeCorrection/15SVD_20091211_S2A_L1Csimu_aot.tif
- dtm DTM/SENTINEL2/VE_TEST_AUX_REFDE2_15SDV____0001.DBL.DIR/S2N_TEST_AUX_REFDE2_15SVD_0001_AI
- edg SlopeCorrection/15SVD_20091211_S2A_L1Csimu_edg.tif
- thetas 64.1652

6.5.23.4 Liste des produits de sortie

- tdir TVA/15SVD_20091211_S2A_L1Csimu_tdir.tif
- tdif TVA/15SVD_20091211_S2A_L1Csimu_tdif.tif

6.5.23.5 Prérequis

Il n'y a pas de prérequis.

6.5.23.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.23.7 Epsilon utilisé sur la non regression

0

6.5.23.8 Vérifications à effectuer

Le test génère en sortie les images de transmission tdir et tdif pour la correction des effets de pentes.

6.5.23.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvComputeTransmission

6.5.23.10 Journal d'essai de la recette

6.5.23.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.23.10.2 Conclusion du déroulement du test

RAS

6.5.23.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.23.11 Exigences

Ce test couvre les exigences suivantes :

6.5.24 apTvConcatenatePerZone

6.5.24.1 Objectif

Validation de l'application "ConcatenatePerZone"

6.5.24.2 Description

Le module "ConcatenatePerZone" concatène plusieurs images ayant chacune un identifiant de zone en fonction d'un masque de zone.

6.5.24.3 Liste des données d'entrées

- Plusieurs images représentant chacune une zone (par exemple images d'angle pour chaque détecteur Sentinel2).
- Masque de zone

6.5.24.4 Liste des produits de sortie

Image dont le contenu est la répartition selon le masque des images d'entrée. Typiquement un assemblage des différents détecteur.

6.5.24.5 Prérequis

Il n'y a pas de prérequis.

6.5.24.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.24.7 Epsilon utilisé sur la non regression

0

6.5.24.8 Vérifications à effectuer

Le test génère en sortie une image dont la répartition est selon le masque des images d'entrée.

6.5.24.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvConcatenatePerZone

6.5.24.10 Journal d'essai de la recette

6.5.24.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.24.10.2 Conclusion du déroulement du test

RAS

6.5.24.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.24.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.25 apTvDarkestPixelAOTNominal

6.5.25.1 Objectif

Validation de l'application "DarkestPixelAOTNominal" calcul la valeur du DarkestPixel pour l'algorithme d'AOTEstimation

6.5.25.2 Description

Le module "DarkestPixelAOTNominal" calcul la valeur du DarkestPixel pour l'algorithme d'AOTEstimation .

6.5.25.3 Liste des données d'entrées

- toac AOTEstimation/L930662_20020212_L7_198_030_USGS_toac_240m.tif
- shd AOTEstimation/L930662_20020212_L7_198_030_USGS_shd_240m.tif
- dtm DTM/LANDSAT/LN_TEST_AUX_REFDE2_EU93066200A00B_0001.DBL.DIR/LN_TEST_AUX_REFDE2_EU93066200A00B_0001.DBL
- cld AOTEstimation/L930662_20020212_L7_198_030_USGS_l2cld_240m.tif
- l2ndt AOTEstimation/L930662_20020212_L7_198_030_USGS_l2ndt_240m.tif
- l2rtc AOTEstimation/L930662_20020212_L7_198_030_USGS_l2rtc_240m.tif
- edg AOTEstimation/L930662_20020212_L7_198_030_USGS_edg_240m.tif

- lutheader AOTestimation/giplut.xml
- lutimage AOTestimation/lutAOT.mha
- darkestchanneltoac 0
- thresholddarkpixel 0.015
- darksurfacereflectance 0
- darkestchannelrtc 0

6.5.25.4 Liste des produits de sortie

Valeurs du DarkestPixel

6.5.25.5 Prérequis

Il n'y a pas de prérequis.

6.5.25.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.25.7 Epsilon utilisé sur la non regression

0

6.5.25.8 Vérifications à effectuer

Le test génère en sortie la valeurs du DarkestPixel.

6.5.25.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvDarkestPixelAOTNominal

6.5.25.10 Journal d'essai de la recette

6.5.25.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.25.10.2 Conclusion du déroulement du test

RAS

6.5.25.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.25.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.26 apTvDirectionalCorrectionROY

6.5.26.1 Objectif

Validation de l'application "DirectionalCorrection" avec le modèle ROY.

6.5.26.2 Description

Le module "DirectionalCorrection" calcule les coefficients de correction directionnelle (modèle ROY).

6.5.26.3 Liste des données d'entrées

- données d'orientation

6.5.26.4 Liste des produits de sortie

Valeurs des coefficients de correction

6.5.26.5 Prérequis

Il n'y a pas de prérequis.

6.5.26.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.26.7 Epsilon utilisé sur la non regression

0

6.5.26.8 Vérifications à effectuer

Les valeurs de corrections doivent être crédible par rapport à l'orientation du satellite

6.5.26.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvDirectionalCorrectionROY

6.5.26.10 Journal d'essai de la recette

6.5.26.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.26.10.2 Conclusion du déroulement du test

RAS

6.5.26.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.26.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.27 apTvDirectionalCorrectionLUT

6.5.27.1 Objectif

Validation de l'application "DirectionalCorrection" avec le modèle LUT.

6.5.27.2 Description

Le module "DirectionalCorrection" calcule les coefficients de correction directionnelle (modèle LUT).

6.5.27.3 Liste des données d'entrées

- données d'orientation

6.5.27.4 Liste des produits de sortie

Valeurs des coefficients de correction

6.5.27.5 Prérequis

Il n'y a pas de prérequis.

6.5.27.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.27.7 Epsilon utilisé sur la non regression

0

6.5.27.8 Vérifications à effectuer

Les valeurs de corrections doivent être crédible par rapport à l'orientation du satellite

6.5.27.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : `cctest -R apTvDirectionalCorrectionLUT`

6.5.27.10 Journal d'essai de la recette

6.5.27.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.27.10.2 Conclusion du déroulement du test

RAS

6.5.27.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.27.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.28 apTvDispatchZoneToVector

6.5.28.1 Objectif

Validation de l'application "DispatchZoneToVector" avec des images conforme en entrée

6.5.28.2 Description

Le module "DispatchZoneToVector" déconcatène plusieurs images contenant chacune une zone mais plusieurs bande en plusieurs bandes contenant chacune leurs zones respectives. Cette application est spécifique au masque DETF de Sentinel2 format muscate. Cette version du test considère que les bande suivent correctement l'ordre, il ne manque pas de bande pour un détecteur donné.

6.5.28.3 Liste des données d'entrées

- Plusieurs images définissant les zones mais contenant chacune plusieurs bandes répartie sur plusieurs bit, par exemple un fichier S2_Muscate : SENTINEL2B_20171008-105012-463_L2A_T31TCH_C_V1-0_DTF_R3-D06.tif.

6.5.28.4 Liste des produits de sortie

Image dont chaque bande contiens ses zones couverte (DETFOO par exemple).

6.5.28.5 Prérequis

Il n'y a pas de prérequis.

6.5.28.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.28.7 Epsilon utilisé sur la non regression

0

6.5.28.8 Vérifications à effectuer

Le test génère en sortie une image dont la répartition est selon les images de zone d'entrée pour chaque bande.

6.5.28.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvDispatchZoneToVector

6.5.28.10 Journal d'essai de la recette

6.5.28.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.28.10.2 Conclusion du déroulement du test

RAS

6.5.28.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.28.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.29 apTvDispatchZoneToVectorWeight

6.5.29.1 Objectif

Validation de l'application "DispatchZoneToVector" avec des images non conforme en entrée

6.5.29.2 Description

Le module “DispatchZoneToVector” déconcatène plusieurs images contenant chacune une zone mais plusieurs bande en plusieurs bandes contenant chacune leurs zones respectives. Cette application est spécifique au masque DETF de Sentinel2 format muscate. Cette version du test considère que les bande ne suivent pas correctement l'ordre, il manque des bande pour un détecteur donné. Il arrive qu'une certaine bande ne touche pas un détecteur.

6.5.29.3 Liste des données d'entrées

- Plusieurs images définissant les zones mais contenant chacune plusieurs bandes répartie sur plusieurs bit, par exemple un fichier S2_Muscate : SENTINEL2B_20171008-105012-463_L2A_T31TCH_C_V1-0_DTF_R3-D06.tif. Les images n'ont pas la bonne liste de bande et les bits ne sont pas dans le bon ordre.

6.5.29.4 Liste des produits de sortie

Image dont chaque bande contient ses zones couverte (DETFOO par exemple). Une des bandes ne contient pas tous le détecteur car non présente sur celui-ci

6.5.29.5 Prérequis

Il n'y a pas de prérequis.

6.5.29.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.29.7 Epsilon utilisé sur la non regression

0

6.5.29.8 Vérifications à effectuer

Le test génère en sortie une image dont la répartition est selon les images de zone d'entrée pour chaque bande. Une des bandes ne contient pas tous le détecteur car non présente sur celui-ci

6.5.29.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : `ctest -R apTvDispatchZoneToVectorWeight`

6.5.29.10 Journal d'essai de la recette

6.5.29.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.29.10.2 Conclusion du déroulement du test

RAS

6.5.29.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.29.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.30 apTvDTMProcessing

6.5.30.1 Objectif

Validation de l'application "DTMProcessing"

6.5.30.2 Description

Le module "DTMProcessing" calcul les parties cachées et les ombre du DTM sur le produit.

Le module "DTMProcessing" détecte les pixels correspondant aux faces cachées par le relief et aux zones d'ombres liées également au relief. L'image utilisée pour ce test est une image située sur le site Maroc qui présente de légères variations du relief avec des versants NORD à l'ombre. Pour accentuer cet effet, les angles d'acquisition sont modifiés

6.5.30.3 Liste des données d'entrées

- solar DTMProcessingTest/Maroc_20061216_solar.tif
- viedtm DTMProcessingTest/Maroc_20061216_vie.tif
- edg DTMProcessingTest/Maroc_20061216_edg.tif
- dtm DTM/FORMOSAT/VE_TEST_AUX_REFDE2_MAROC____0001.DBL.DIR/Maroc_96m.mnt_ALC
- distthreshold 96
- solhref 4000
- viehref 4000

6.5.30.4 Liste des produits de sortie

- hid TVA/apTvDTMProcessing_hid.tif
- shd TVA/apTvDTMProcessing_shd.tif

6.5.30.5 Prérequis

Il n'y a pas de prérequis.

6.5.30.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.30.7 Epsilon utilisé sur la non regression

0

6.5.30.8 Vérifications à effectuer

Le test génère en sortie les images composites

6.5.30.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvDTMProcessing

6.5.30.10 Journal d'essai de la recette

6.5.30.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.30.10.2 Conclusion du déroulement du test

RAS

6.5.30.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.30.11 Exigences

Ce test couvre les exigences suivantes : ST-169-DTM-0010 (C) ; ST-169-DTM-0020 (C) ; ST-169-DTM-0030 (C) ; ST-169-DTM-0040(C) ; ST-169-DTM-0050 (C) ;

6.5.31 apTvEnvCorrection

6.5.31.1 Objectif

Validation de l'application "EnvCorrection"

6.5.31.2 Description

Le test "EnvCorrection" applique la correction environnementale à une image.

Le module EnvironmentCorrection corrige l'image de réflectance de surface (issue de "Scattering Correction") des effets de l'environnement. Cette correction s'appuie sur les LUT Tdir, Tdif et Satm ainsi que sur le MNT fourni en entrée.

6.5.31.3 Liste des données d'entrées

- tdir EnvCorrection/Sudouest_20060329_MS_fmsat_ortho_surf_96m_Tdir.tif
- tdif EnvCorrection/Sudouest_20060329_MS_fmsat_ortho_surf_96m_Tdif.tif
- albd EnvCorrection/Sudouest_20060329_MS_fmsat_ortho_surf_96m_Satm.tif
- toc EnvCorrection/Sudouest_20060329_MS_fmsat_ortho_surf_96m_toc.tif
- edg EnvCorrection/Sudouest_20060329_MS_fmsat_ortho_surf_96m_edg.tif
- rhoenvsub apTvComputeRhoEnv.tif
- nodata -10

6.5.31.4 Liste des produits de sortie

- sre \${MAJA_TEST_OUTPUT_ROOT}/TVA/apTvEnvCorrection_SRE.tif
- rhoenv \${MAJA_TEST_OUTPUT_ROOT}/TVA/apTvEnvCorrection_RhoEnv.tif

6.5.31.5 Prérequis

Il n'y a pas de prérequis.

6.5.31.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.31.7 Epsilon utilisé sur la non regression

0

6.5.31.8 Vérifications à effectuer

Le test génère en sortie l'image corrigée.

6.5.31.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvEnvCorrection

6.5.31.10 Journal d'essai de la recette

6.5.31.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.31.10.2 Conclusion du déroulement du test

RAS

6.5.31.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.31.11 Exigences

Ce test couvre les exigences suivantes : ST-169-EnvironmentCorr-0010(C) ;ST-169-EnvironmentCorr-0020(C) ;ST-169-EnvironmentCorr-0030 (C) ; ST-169-EnvironmentCorr-0040 (C) ; ST-169-EnvironmentCorr-0050(C) ;

6.5.32 apTvExtractLutChannel

6.5.32.1 Objectif

Validation de l'application "ExtractLutChannel"

6.5.32.2 Description

Le module "ExtractLutChannel" extrait une bande d'un fichier de lut mha.

6.5.32.3 Liste des données d'entrées

- Fichier lut
- Channel voulue

6.5.32.4 Liste des produits de sortie

Lut à une bande

6.5.32.5 Prérequis

Il n'y a pas de prérequis.

6.5.32.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.32.7 Epsilon utilisé sur la non regression

0

6.5.32.8 Vérifications à effectuer

La lut de sortie doit correspondre à la bande voulue

6.5.32.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvExtractLutChannel

6.5.32.10 Journal d'essai de la recette

6.5.32.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.32.10.2 Conclusion du déroulement du test

RAS

6.5.32.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.32.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.33 apTvGammaCompute

6.5.33.1 Objectif

Validation de l'application "GammaCompute"

6.5.33.2 Description

Le test "GammaCompute" calcul le gamma d'une liste d'image pour la correction cirrus.

6.5.33.3 Liste des données d'entrées

- l2toa CirrusCorrectionTest/11SLT_20150803_S2A_L1Csimu_toa_abs_10m.tiff
- l2edg CirrusCorrectionTest/11SLT_20150803_S2A_L1Csimu_bord_10m.bord_final
- l2dtm DTM/SENTINEL2/11SLT/11SLT_10m.mnt
- l2cirrus CirrusCorrectionTest/11SLT_20150803_S2A_L1Csimu_toa_60m.B10_resampled.tif
- bandlist 2 3
- cloudcirrus CirrusCorrectionTest/11SLT_20150803_S2A_L1Csimu_cloud_cirrus_10m.tif
- cloudall CirrusCorrectionTest/11SLT_20150803_S2A_L1Csimu_all_mask_10m.tif
- cloudrefl CirrusCorrectionTest/11SLT_20150803_S2A_L1Csimu_refl_mask_10m.tif
- cirrusminpixinlist 100

6.5.33.4 Liste des produits de sortie

Le test génère la valeur du gamma ainsi que le mask de cirrus rééchantillonné

6.5.33.5 Prérequis

Il n'y a pas de prérequis.

6.5.33.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.33.7 Epsilon utilisé sur la non regression

0

6.5.33.8 Vérifications à effectuer

Le test génère en sortie une valeur de gamma ainsi que le mask de cirrus rééchantillonné

6.5.33.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvGammaCompute

6.5.33.10 Journal d'essai de la recette

6.5.33.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.33.10.2 Conclusion du déroulement du test

RAS

6.5.33.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.33.11 Exigences

Ce test couvre les exigences suivantes : Néan

6.5.34 apTvGapFilling

6.5.34.1 Objectif

Validation de l'application "GapFilling" qui applique l'algo du GapFilling de l'algorithme d'AOTEstimation

6.5.34.2 Description

Le module "GapFilling" applique l'algo du GapFilling de l'algorithme d'AOTEstimation

6.5.34.3 Liste des données d'entrées

- im AOTEstimation/L930662_20020212_L7_198_030_USGS_aot.tif
- initwindowsradius 2
- maxwindowsradius 16
- reall2nodata -10
- mean 0.02948
- defaultvalue 0.2
- hasvalidpixels 1

6.5.34.4 Liste des produits de sortie

- out GapFilling_filledImage.tif
- outmask GapFilling_mask.tif

6.5.34.5 Prérequis

Il n'y a pas de prérequis.

6.5.34.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.34.7 Epsilon utilisé sur la non regression

0

6.5.34.8 Vérifications à effectuer

Le test génère en sortie l'image GapFilled ainsi que le masque de GapFilling.

6.5.34.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvGapFilling

6.5.34.10 Journal d'essai de la recette

6.5.34.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.34.10.2 Conclusion du déroulement du test

RAS

6.5.34.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.34.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.35 apTvImageListWriter

6.5.35.1 Objectif

Validation de l'application "ImageListWriter"

6.5.35.2 Description

Le module "ImageListWriter" écrit simultanément une liste d'image.

6.5.35.3 Liste des données d'entrées

Liste d'images

6.5.35.4 Liste des produits de sortie

Liste d'images

6.5.35.5 Prérequis

Il n'y a pas de prérequis.

6.5.35.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.35.7 Epsilon utilisé sur la non regression

0

6.5.35.8 Vérifications à effectuer

Le test génère en sortie une copie des images d'entrées.

6.5.35.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : `ctest -R apTvImageListWriter`

6.5.35.10 Journal d'essai de la recette

6.5.35.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.35.10.2 Conclusion du déroulement du test

RAS

6.5.35.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.35.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.36 apTvIncidenceAngle**6.5.36.1 Objectif**

Validation de l'application "IncidenceAngle"

6.5.36.2 Description

Le module "IncidenceAngle" calcul les angles d'incidence terrain à partir des angles solaires et des données de pente et d'altitudes du DTM.

6.5.36.3 Liste des données d'entrées

- DEM : altitude et pente (ASP/SPL)
- Angle solaires

6.5.36.4 Liste des produits de sortie

Image contenant les angle d'incidence pour chaque pixel

6.5.36.5 Prérequis

Il n'y a pas de prérequis.

6.5.36.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.36.7 Epsilon utilisé sur la non regression

0

6.5.36.8 Vérifications à effectuer

Le test génère en sortie une image contenant les angles d'incidences.

6.5.36.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvIncidenceAngle

6.5.36.10 Journal d'essai de la recette

6.5.36.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.36.10.2 Conclusion du déroulement du test

RAS

6.5.36.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.36.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.37 apTvInterpolateForEnvCor

6.5.37.1 Objectif

Validation de l'application "InterpolateForEnvCor"

6.5.37.2 Description

Le module "InterpolateForEnvCor" calcul les lut de transmission pour la correction environnementale.

6.5.37.3 Liste des données d'entrées

- lutalbd EnvCorrection/giplutalbedomap.xml
- luttdir SlopeCorrection/giplutTdirmap.xml
- luttdif SlopeCorrection/giplutTdifmap.xml
- aot SlopeCorrection/15SVD_20091211_S2A_L1Csimu_aot.tif
- dtm DTM/SENTINEL2/VE_TEST_AUX_REFDE2_15SDV____0001.DBL.DIR/S2N_TEST_AUX_REFDE2_15SVD_0001_AI
- edg SlopeCorrection/15SVD_20091211_S2A_L1Csimu_edg.tif
- thetav 0 0 0 0 0 0 0 0 0 0 0 0

6.5.37.4 Liste des produits de sortie

- tdir TVA/apTvInterpolateForEnvCorr_Tdir.tif
- tdif TVA/apTvInterpolateForEnvCorr_Tdif.tif
- albd TVA/apTvInterpolateForEnvCorr_albedo.tif

6.5.37.5 Prérequis

Il n'y a pas de prérequis.

6.5.37.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.37.7 Epsilon utilisé sur la non regression

0

6.5.37.8 Vérifications à effectuer

Le test génère en sortie les images de lut interolées

6.5.37.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvInterpolateForEnvCor

6.5.37.10 Journal d'essai de la recette

6.5.37.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.37.10.2 Conclusion du déroulement du test

RAS

6.5.37.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.37.11 Exigences

Ce test couvre les exigences suivantes :

6.5.38 apTvMultiplyByScalar

6.5.38.1 Objectif

Validation de l'application "MultiplyByScalar"

6.5.38.2 Description

Le module "MultiplyByScalar" multiplie l'image par un scalaire.

6.5.38.3 Liste des données d'entrées

Image lambda

6.5.38.4 Liste des produits de sortie

Image lambda multipliée

6.5.38.5 Prérequis

Il n'y a pas de prérequis.

6.5.38.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.38.7 Epsilon utilisé sur la non regression

0

6.5.38.8 Vérifications à effectuer

Le test génère en sortie une image multipliée à l'originale.

6.5.38.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvMultiplyByScalar

6.5.38.10 Journal d'essai de la recette

6.5.38.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.38.10.2 Conclusion du déroulement du test

RAS

6.5.38.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.38.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.39 apTvNewCamsCompute

6.5.39.1 Objectif

Validation de l'application "CamsCompute" avec des CAMS nouveau format

6.5.39.2 Description

Le module "CamsCompute" extrait les valeurs de pondération cams d'un fichier CAMS.

6.5.39.3 Liste des données d'entrées

- Fichier CAMS nouvelle version
- Lat/lon souhaité

6.5.39.4 Liste des produits de sortie

Valeurs de pondération des modèles pour la localisation demandée

6.5.39.5 Prérequis

Il n'y a pas de prérequis.

6.5.39.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.39.7 Epsilon utilisé sur la non regression

0

6.5.39.8 Vérifications à effectuer

Les valeurs de pondération doivent correspondre avec la zone

6.5.39.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvNewCamsCompute

6.5.39.10 Journal d'essai de la recette

6.5.39.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.39.10.2 Conclusion du déroulement du test

RAS

6.5.39.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.39.11 Exigences

Ce test couvre les exigences suivantes : ST-169-ReducedLUT-0010 (C) ;

6.5.40 apTvNewCamsMixedCompute

6.5.40.1 Objectif

Validation de l'application "CamsCompute" avec des CAMS nouveau format et des anciens format en même temps

6.5.40.2 Description

Le module "CamsCompute" extrait les valeurs de pondération cams d'un fichier CAMS.

6.5.40.3 Liste des données d'entrées

- Fichier CAMS nouvelle version et ancienne version
- Lat/lon souhaité

6.5.40.4 Liste des produits de sortie

Valeurs de pondération des modèles pour la localisation demandée

6.5.40.5 Prérequis

Il n'y a pas de prérequis.

6.5.40.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.40.7 Epsilon utilisé sur la non regression

0

6.5.40.8 Vérifications à effectuer

Les valeurs de pondération doivent correspondre avec la zone

6.5.40.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvNewCamsMixedCompute

6.5.40.10 Journal d'essai de la recette

6.5.40.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.40.10.2 Conclusion du déroulement du test

RAS

6.5.40.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.40.11 Exigences

Ce test couvre les exigences suivantes : ST-169-ReducedLUT-0010 (C) ;

6.5.41 apTvOneBandEqualThreshold

6.5.41.1 Objectif

Validation de l'application "OneBandEqualThreshold"

6.5.41.2 Description

Le module "OneBandEqualThreshold" génère un masque dont la valeur vaut 'inside_value' si au moins une des bande vaut 'threshold', 'outside_value' sinon

6.5.41.3 Liste des données d'entrées

Image à plusieurs bande

6.5.41.4 Liste des produits de sortie

Masque

6.5.41.5 Prérequis

Il n'y a pas de prérequis.

6.5.41.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.41.7 Epsilon utilisé sur la non regression

0

6.5.41.8 Vérifications à effectuer

Le test génère en sortie une image arrondies à plusieurs bandes.

6.5.41.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvOneBandEqualThreshold

6.5.41.10 Journal d'essai de la recette

6.5.41.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.41.10.2 Conclusion du déroulement du test

RAS

6.5.41.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.41.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.42 apTvOzoneExtractor

6.5.42.1 Objectif

Validation de l'application "OzoneExtractor"

6.5.42.2 Description

Le module "OzoneExtractor" extrait la valeur d'ozone d'un fichier météo.

6.5.42.3 Liste des données d'entrées

- Fichier météo
- Lat/lon souhaité

6.5.42.4 Liste des produits de sortie

Valeur d'ozone pour la localisation demandée

6.5.42.5 Prérequis

Il n'y a pas de prérequis.

6.5.42.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.42.7 Epsilon utilisé sur la non regression

0

6.5.42.8 Vérifications à effectuer

La valeurs d'ozone doit correspondre avec le fichier d'entrée (attention aux conversions)

6.5.42.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvOzoneExtractor

6.5.42.10 Journal d'essai de la recette

6.5.42.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.42.10.2 Conclusion du déroulement du test

RAS

6.5.42.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.42.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.43 apTvPressure

6.5.43.1 Objectif

Validation de l'application "Pressure"

6.5.43.2 Description

Le test "Pressure" calcul la valeur de la pression atmosphérique en fonction de l'altitude.

6.5.43.3 Liste des données d'entrées

- dtm DTM/FORMOSAT/VE_TEST_AUX_REFDE2_SUDOUEST_0001.DBL.DIR/Sudouest_96m.mnt_ALC

6.5.43.4 Liste des produits de sortie

- pressure TVA/aptvAthmosphericAbsorption_PressureFilter.tif

6.5.43.5 Prérequis

Il n'y a pas de prérequis.

6.5.43.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.43.7 Epsilon utilisé sur la non regression

0

6.5.43.8 Vérifications à effectuer

Le test génère en sortie l'image des valeurs de pression.

6.5.43.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvPressure

6.5.43.10 Journal d'essai de la recette

6.5.43.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.43.10.2 Conclusion du déroulement du test

RAS

6.5.43.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.43.11 Exigences

Ce test couvre les exigences suivantes : Néan

6.5.44 apTvRayleighCorrection

6.5.44.1 Objectif

Validation de l'application "RayleighCorrection"

6.5.44.2 Description

Le test "RayleighCorrection" applique la correction rayleigh à une image.

Le module RayleighCorrection corrige les images de réflectance TOA des effets de la diffusion liés aux molécules et aux aérosols. L'estimation du contenu en aérosols n'ayant pas encore été réalisée, une valeur faible d'épaisseur optique aérosol (voire nulle ici) est fixée par défaut

6.5.44.3 Liste des données d'entrées

- lutimage RayleighCorrection/lut.mha
- lutheader RayleighCorrection/giplut.xml
- toa RayleighCorrection/Sudouest_20060819_MS_fmsat_ortho_toa_96m_AOTnull.tif
- edg RayleighCorrection/Sudouest_20060819_MS_fmsat_ortho_toa_bord_96m
- dtm DTM/FORMOSAT/VE_TEST_AUX_REFDE2_SUDOUEST_0001.DBL.DIR/Sudouest_96m.mnt_ALC
- nodata -10

- defaultaot 0

6.5.44.4 Liste des produits de sortie

- tocr TVA/apTvRayleighCorrection.tif

6.5.44.5 Prérequis

Il n'y a pas de prérequis.

6.5.44.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.44.7 Epsilon utilisé sur la non regression

0

6.5.44.8 Vérifications à effectuer

Le test génère en sortie l'image corrigée.

6.5.44.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvRayleighCorrection

6.5.44.10 Journal d'essai de la recette

6.5.44.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.44.10.2 Conclusion du déroulement du test

RAS

6.5.44.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.44.11 Exigences

Ce test couvre les exigences suivantes : ST-169-RayleighCorr-0010 (C) ;

6.5.45 apTvReduceLutDirCorr

6.5.45.1 Objectif

Validation de l'application "ReduceLut" en appliquant la correction directionnelle

6.5.45.2 Description

Le module "ReduceLut" extrait une lut 3D à partir d'une lut 6D et des angles de vues du détecteur. De plus une correction directionnelle est appliquée dans le cas d'un capteur ayant les données nécessaires

6.5.45.3 Liste des données d'entrées

- solar.azimuth 143.195
- solar.zenith 24.0717
- viewing.zenith 6.08321 6.08321 6.08321 6.08321 6.08321 6.08321 6.08321 6.08321 6.08321 6.08321 6.08321 6.08321 6.08321 6.08321
- viewing.azimuth 104.105 104.105 104.105 104.105 104.105 104.105 104.105 104.105 104.105 104.105 104.105 104.105 104.105 104.105
- lut ReduceLutComputationTest/giplut_cams.xml
- refzenith 0
- refazimuth 0
- dircorr true
- coeffsr 0 0.480620155 0.4441041348 0.3396449704 0.3396449704 0.4179634487 0.4962819269 0.4962819269 0.4962819269 0 0 0.3364431487 0.240406320
- coeffsv 0 0.1020671835 0.1362940276 0.1343195266 0.1343195266 0.1205060291 0.1066925315 0.1066925315 0.1066925315 0 0 0.1320699708 0.1455981941

6.5.45.4 Liste des produits de sortie

- reducelut TVA/apTvReduceLutDirCorr.mhd

6.5.45.5 Prérequis

Il n'y a pas de prérequis.

6.5.45.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.45.7 Epsilon utilisé sur la non regression

0

6.5.45.8 Vérifications à effectuer

La lut 3D doit correspondre à la lut6D extraite sur les paramètres

6.5.45.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvReduceLutDirCorr

6.5.45.10 Journal d'essai de la recette

6.5.45.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.45.10.2 Conclusion du déroulement du test

RAS

6.5.45.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.45.11 Exigences

Ce test couvre les exigences suivantes : ST-169-ReducedLUT-0010 (C) ;

6.5.46 apTvReduceLut

6.5.46.1 Objectif

Validation de l'application "ReduceLut"

6.5.46.2 Description

Le module "ReduceLut" extrait une lut 3D à partir d'une lut 6D et des angles de vues du détecteur.

6.5.46.3 Liste des données d'entrées

- solar.azimuth 143.195
- solar.zenith 24.0717
- viewing.zenith 6.08321 6.08321 6.08321 6.08321 6.08321 6.08321 6.08321 6.08321 6.08321 6.08321 6.08321 6.08321 6.08321 6.08321
- viewing.azimuth 104.105 104.105 104.105 104.105 104.105 104.105 104.105 104.105 104.105 104.105 104.105 104.105 104.105 104.105
- lut ReduceLutComputationTest/giplut_cams.xml

6.5.46.4 Liste des produits de sortie

- reducelut TVA/apTvReduceLut.mhd

6.5.46.5 Prérequis

Il n'y a pas de prérequis.

6.5.46.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.46.7 Epsilon utilisé sur la non regression

0

6.5.46.8 Vérifications à effectuer

La lut 3D doit correspondre à la lut6D extraite sur les paramètres

6.5.46.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvReduceLut

6.5.46.10 Journal d'essai de la recette

6.5.46.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.46.10.2 Conclusion du déroulement du test

RAS

6.5.46.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.46.11 Exigences

Ce test couvre les exigences suivantes : ST-169-ReducedLUT-0010 (C) ;

6.5.47 apTvResampling

6.5.47.1 Objectif

Validation de l'application "Resampling"

6.5.47.2 Description

Le module "Resampling" rééchantillonne une image sur un DTM donné.

6.5.47.3 Liste des données d'entrées

- Image lambda
- DTM coarse

6.5.47.4 Liste des produits de sortie

Image lambda reéchantillonnée sur l'emprise du DTM

6.5.47.5 Prérequis

Il n'y a pas de prérequis.

6.5.47.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.47.7 Epsilon utilisé sur la non regression

0

6.5.47.8 Vérifications à effectuer

Le test génère en sortie une image dont l'emprise est calée sur le DTM

6.5.47.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvResampling

6.5.47.10 Journal d'essai de la recette

6.5.47.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.47.10.2 Conclusion du déroulement du test

RAS

6.5.47.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.47.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.48 apTvRescaleIntensity

6.5.48.1 Objectif

Validation de l'application "RescaleIntensity"

6.5.48.2 Description

Le module "RescaleIntensity" modifie la dynamique de l'image.

6.5.48.3 Liste des données d'entrées

Image lambda

6.5.48.4 Liste des produits de sortie

Image lambda ayant vu sa dynamique modifiée

6.5.48.5 Prérequis

Il n'y a pas de prérequis.

6.5.48.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.48.7 Epsilon utilisé sur la non regression

0

6.5.48.8 Vérifications à effectuer

Le test génère en sortie une image arrondies à plusieurs bandes.

6.5.48.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvRescaleIntensity

6.5.48.10 Journal d'essai de la recette

6.5.48.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.48.10.2 Conclusion du déroulement du test

RAS

6.5.48.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.48.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.49 apTvRoundImage

6.5.49.1 Objectif

Validation de l'application "RoundImage"

6.5.49.2 Description

Le module "RoundImage" fait un arrondis d'une image flottante.

6.5.49.3 Liste des données d'entrées

Image de préférence en valeurs flottantes

6.5.49.4 Liste des produits de sortie

Image arrondies à l'entier le plus proche

6.5.49.5 Prérequis

Il n'y a pas de prérequis.

6.5.49.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.49.7 Epsilon utilisé sur la non regression

0

6.5.49.8 Vérifications à effectuer

Le test génère en sortie une image arrondies.

6.5.49.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvRoundImage

6.5.49.10 Journal d'essai de la recette

6.5.49.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.49.10.2 Conclusion du déroulement du test

RAS

6.5.49.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.49.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.50 apTvRoundVectorImage

6.5.50.1 Objectif

Validation de l'application "RoundVectorImage"

6.5.50.2 Description

Le module "RoundImage" fait un arrondis d'une image flottante à plusieurs bandes.

6.5.50.3 Liste des données d'entrées

Image de préférence en valeurs flottantes à plusieurs bandes

6.5.50.4 Liste des produits de sortie

Image arrondies à l'entier le plus proche à plusieurs bandes

6.5.50.5 Prérequis

Il n'y a pas de prérequis.

6.5.50.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.50.7 Epsilon utilisé sur la non regression

0

6.5.50.8 Vérifications à effectuer

Le test génère en sortie une image arrondies à plusieurs bandes.

6.5.50.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvRoundVectorImage

6.5.50.10 Journal d'essai de la recette

6.5.50.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.50.10.2 Conclusion du déroulement du test

RAS

6.5.50.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.50.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.51 apTvScatteringCorrection

6.5.51.1 Objectif

Validation de l'application "ScatteringCorrection"

6.5.51.2 Description

Le test "ScatteringCorrection" applique la correction du scattering à une image. Le module ScatteringCorrection corrige les images de réflectance des effets de la diffusion liée aux molécules et aux aérosols. Une image du contenu en aérosols est fournie en entrée.

6.5.51.3 Liste des données d'entrées

- lutmap ScatteringCorrectionTest/giplutmap.xml
- toac ScatteringCorrectionTest/apTvSudouest_20060819_toac.tif
- aot ScatteringCorrectionTest/apTvSudouest_20060819_aot.tif
- edg ScatteringCorrectionTest/apTvSudouest_20060819_edg.tif
- dtm ScatteringCorrectionTest/apTvSudouest_20060819_dtm.tif
- l2nodata 0
- l2bandincoarse 0 1 2 3

6.5.51.4 Liste des produits de sortie

- tocr TVA/apTvScatteringCorrectionImageFilterTest.tif

6.5.51.5 Prérequis

Il n'y a pas de prérequis.

6.5.51.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.51.7 Epsilon utilisé sur la non regression

0

6.5.51.8 Vérifications à effectuer

Le test génère en sortie l'image corrigée.

6.5.51.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvScatteringCorrection

6.5.51.10 Journal d'essai de la recette

6.5.51.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.51.10.2 Conclusion du déroulement du test

RAS

6.5.51.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.51.11 Exigences

Ce test couvre les exigences suivantes : ST-169-SactteringCorr-0010 (C) ;

6.5.52 apTvSlopeCorrection

6.5.52.1 Objectif

Validation de l'application "SlopeCorrection"

6.5.52.2 Description

Le test "SlopeCorrection" applique la correction des effets de pentes à une image.

Le module SlopeCorrection corrige radiométriquement l'image de reflectance de surface corrigée des effets de l'environnement (issue de "EnvironmentCorrection") des effets liés à la topographie. Cette correction s'appuie sur les images de pente, d'aspect et de MNT fournies en entrée, ainsi que sur les LUT Dirt, Dift et Satm.

6.5.52.3 Liste des données d'entrées

- tdirsub SlopeCorrection/15SVD_20091211_S2A_L1Csimu_tdir.tif
- tdifsub SlopeCorrection/15SVD_20091211_S2A_L1Csimu_tdif.tif
- dtmasc DTM/SENTINEL2/VE_TEST_AUX_REFDE2_15SDV____0001.DBL.DIR/S2N_TEST_AUX_REFDE2_15SVD_0001
- dtmslc DTM/SENTINEL2/VE_TEST_AUX_REFDE2_15SDV____0001.DBL.DIR/S2N_TEST_AUX_REFDE2_15SVD_0001
- rhoenv SlopeCorrection/15SVD_20091211_S2A_L1Csimu_rho_env_10m.tif
- incangle SlopeCorrection/15SVD_20091211_S2A_L1Csimu_IncidenceAngle.tif
- sre SlopeCorrection/15SVD_20091211_S2A_L1Csimu_surf_env_10m.tif
- thetas 64.165163327
- phis 162.402228733
- thetav 0 0 0 0 0 0
- phiv 0 0 0 0 0 0
- mincosi 0.2000000000000000011102230246252
- mincose 0.1000000000000000005551115123126
- mincosratio 0.2000000000000000011102230246252
- nodata -10

6.5.52.4 Liste des produits de sortie

- fre TVA/apTvSlopeCorrection_15SVD_20091211_S2A_fre.tif
- tgs TVA/apTvSlopeCorrection_15SVD_20091211_S2A_tgs.tif
- stl TVA/apTvSlopeCorrection_15SVD_20091211_S2A_stl.tif

6.5.52.5 Prérequis

Il n'y a pas de prérequis.

6.5.52.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.52.7 Epsilon utilisé sur la non regression

0

6.5.52.8 Vérifications à effectuer

Le test génère en sortie l'image corrigée.

6.5.52.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvSlopeCorrection

6.5.52.10 Journal d'essai de la recette

6.5.52.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.52.10.2 Conclusion du déroulement du test

RAS

6.5.52.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.52.11 Exigences

Ce test couvre les exigences suivantes : ST-169-SlopeCorr-0010 (C) ; ST-169-SlopeCorr-0020 (C) ; ST-169-SlopeCorr-0030 (C) ; ST-169-SlopeCorr-0040 (C) ; ST-169-SlopeCorr-0050 (C) ; ST-169-SlopeCorr-0060 (C) ;

6.5.53 apTvSmoothingBis

6.5.53.1 Objectif

Validation de l'application "SmoothingBis"

6.5.53.2 Description

Le module "Smoothing" applique un opérateur gaussien sur l'image.

6.5.53.3 Liste des données d'entrées

Image lambda

6.5.53.4 Liste des produits de sortie

Image lambda floutée

6.5.53.5 Prérequis

Il n'y a pas de prérequis.

6.5.53.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.53.7 Epsilon utilisé sur la non regression

0

6.5.53.8 Vérifications à effectuer

Le test génère en sortie une image floutée par rapport à l'originale.

6.5.53.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvSmoothing

6.5.53.10 Journal d'essai de la recette

6.5.53.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.53.10.2 Conclusion du déroulement du test

RAS

6.5.53.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.53.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.54 apTvSnowMask

6.5.54.1 Objectif

Validation de l'application "SnowMask"

6.5.54.2 Description

Le module "SnowMask" calcul le masque de neige du produit.

6.5.54.3 Liste des données d'entrées

- tocr SnowMaskTest/L930662_20020212_L7_198_030_USGS_tocr.tif
- edg SnowMaskTest/L930662_20020212_L7_198_030_USGS_edg.tif
- angles SnowMaskTest/L930662_20020212_L7_198_030_USGS_angle_image.tif
- refband 1
- absband 4
- redband 2
- seradius 2
- ndsithresh 0.4

- redthresh 0.12
- swirthresh 0.16
- thetas 61.48641010000001
- mincos 0.20

6.5.54.4 Liste des produits de sortie

- mask TVA/apTvL930662_20020212_L7_198_030_USGS_snow_mask.tif

6.5.54.5 Prérequis

Il n'y a pas de prérequis.

6.5.54.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.54.7 Epsilon utilisé sur la non regression

0

6.5.54.8 Vérifications à effectuer

Le test génère en sortie le masque de neige du produit.

6.5.54.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvSnowMask

6.5.54.10 Journal d'essai de la recette

6.5.54.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.54.10.2 Conclusion du déroulement du test

RAS

6.5.54.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.54.11 Exigences

Ce test couvre les exigences suivantes : ST-169-SnowMasking-0010 (C) ;

6.5.55 apTvSnowMaskVenus

6.5.55.1 Objectif

Validation de l'application "SnowMask"

6.5.55.2 Description

Le module "SnowMask" calcul le masque de neige du produit dans le cas Venus (sans bande SWIR).

6.5.55.3 Liste des données d'entrées

6.5.55.4 Liste des produits de sortie

6.5.55.5 Prérequis

Il n'y a pas de prérequis.

6.5.55.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.55.7 Epsilon utilisé sur la non regression

0

6.5.55.8 Vérifications à effectuer

Le test génère en sortie le masque de neige du produit.

6.5.55.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvSnowMaskVenus

6.5.55.10 Journal d'essai de la recette

6.5.55.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.55.10.2 Conclusion du déroulement du test

RAS

6.5.55.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.55.11 Exigences

Ce test couvre les exigences suivantes : ST-169-SnowMasking-0010 (C) ;

6.5.56 apTvSolarAngle

6.5.56.1 Objectif

Validation de l'application "SolarAngle"

6.5.56.2 Description

Le module "SolarAngle" calcul les angles solaires à partir des données capteur.

6.5.56.3 Liste des données d'entrées

- SOL1 : Image des angles solaire capteur
- Hauteurs de référence

6.5.56.4 Liste des produits de sortie

Image contenant les angle solaire pour chaque pixel

6.5.56.5 Prérequis

Il n'y a pas de prérequis.

6.5.56.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.56.7 Epsilon utilisé sur la non regression

0

6.5.56.8 Vérifications à effectuer

Le test génère en sortie une image contenant les angles solaires.

6.5.56.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvSolarAngle

6.5.56.10 Journal d'essai de la recette

6.5.56.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.56.10.2 Conclusion du déroulement du test

RAS

6.5.56.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.56.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.57 apTvStats

6.5.57.1 Objectif

Validation de l'application "Stats"

6.5.57.2 Description

Le module "Stats" calcul des statistiques sur l'image.

6.5.57.3 Liste des données d'entrées

Image lambda

6.5.57.4 Liste des produits de sortie

Statistiques de l'image : moyenne, écart type, nombre de pixels valide

6.5.57.5 Prérequis

Il n'y a pas de prérequis.

6.5.57.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.57.7 Epsilon utilisé sur la non regression

0

6.5.57.8 Vérifications à effectuer

Le test génère en sortie des statistiques en accord avec le contenu de l'image.

6.5.57.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvStat

6.5.57.10 Journal d'essai de la recette

6.5.57.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.57.10.2 Conclusion du déroulement du test

RAS

6.5.57.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.57.11 Exigences

Ce test couvre les exigences suivantes : Néant

6.5.58 apTvTOACorrection

6.5.58.1 Objectif

Validation de l'application "TOACorrection"

6.5.58.2 Description

Le test "TOACorrection" applique la correction TOA à une image. Le module AtmosphericAbsorptionCorrection corrige la réflectance au sommet de l'atmosphère de l'absorption atmosphérique. Ce test utilise des contenus en ozone et en vapeur d'eau fixés par défaut. Ce filtre génère l'image corrigée aux résolutions L2 coarse.

6.5.58.3 Liste des données d'entrées

- toa AtmosphericAbsorption/Sudouest_20060819_MS_fmsat_ortho_toa.tif
- pressure aptvAtmosphericAbsorption_PressureFilter.tif
- smac AtmosphericAbsorption/gipsmacmap.xml
- wateramountdefault 3
- ozoneamount 0.299999999999999988897769753748
- thetas 35.557999999999998294697434176

- thetav 22.487999999999995452526491135 22.487999999999995452526491135
22.487999999999995452526491135 22.487999999999995452526491135
- nodata -10
- bandlist 0 1 2 3

6.5.58.4 Liste des produits de sortie

- toac TVA/apTvAthmosphericAbsorption_TOACorrection.tif

6.5.58.5 Prérequis

Il n'y a pas de prérequis.

6.5.58.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.58.7 Epsilon utilisé sur la non regression

0

6.5.58.8 Vérifications à effectuer

Le test génère en sortie l'image des valeurs de pression.

6.5.58.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvTOACorrection

6.5.58.10 Journal d'essai de la recette

6.5.58.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.58.10.2 Conclusion du déroulement du test

RAS

6.5.58.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.58.11 Exigences

Ce test couvre les exigences suivantes : ST-169-AbsorptionCorr-0010 (C) ; ST-169-AbsorptionCorr-0020 (C) ; ST-169-AbsorptionCorr- 0030 (C) ; ST-169-AbsorptionCorr-0030 (C) ; ST-169-AbsorptionCorr-0040 (C) ; ST-169-AbsorptionCorr-0050 (C) ;

6.5.59 apTvWaterAmountGenerator

6.5.59.1 Objectif

Validation de l'application "WaterAmountGenerator"

6.5.59.2 Description

Le test "WaterAmountGenerator" calcul la valeur de la vapeur d'eau.

6.5.59.3 Liste des données d'entrées

- toa AthmosphericAbsorption/31TFJ_20150908_S2A_L1Csimu_toa_240m.tif
- gipwatv AthmosphericAbsorption/giplutmap.xml
- watervaporreferencebandid 8
- watervaporbandid 9
- vapnodatavalue 0
- watervapormaxthreshold 5
- solarzenith 39.999222
- viewingzenith 6.191545

6.5.59.4 Liste des produits de sortie

- wateramount TVA/aptvAthmosphericAbsorption_WaterAmount.tif

6.5.59.5 Prérequis

Il n'y a pas de prérequis.

6.5.59.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.59.7 Epsilon utilisé sur la non regression

0

6.5.59.8 Vérifications à effectuer

Le test génère en sortie l'image des valeurs de vapeur d'eau.

6.5.59.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvWaterAmountGenerator

6.5.59.10 Journal d'essai de la recette

6.5.59.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.59.10.2 Conclusion du déroulement du test

RAS

6.5.59.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.59.11 Exigences

Ce test couvre les exigences suivantes : Néan

6.5.60 apTvWaterMask

6.5.60.1 Objectif

Validation de l'application "WaterMask"

6.5.60.2 Description

Le module "WaterMask" calcul le masque d'eau du produit.

6.5.60.3 Liste des données d'entrées

- tocr WaterMaskingTest/31TFJ_20150928_S2A_L1Csimu_surf_ray_240m.tif
- edg WaterMaskingTest/31TFJ_20150928_S2A_L1Csimu_toa_bord_240m
- cldsum WaterMaskingTest/31TFJ_20150928_S2A_cld.tif
- dtmshd WaterMaskingTest/31TFJ_20150928_S2A_L1Csimu_toa_240m.shd
- l2rcr WaterMaskingTest/31TFJ_20150918_S2A_L1Csimu_surf_ray_comp_240m.tif
- l2pwa WaterMaskingTest/31TFJ_20150918_S2A_L1Csimu_toa_240m.compt
- l2twa WaterMaskingTest/31TFJ_20150918_S2A_L1Csimu_toa_240m.pos
- sunglintflag true
- demslc WaterMaskingTest/31TFJ_240m_slc.tif
- srtmwat WaterMaskingTest/31TFJ_240m.eau
- redbandtocr 3
- nirbandtocr 8
- nirbandrcr 8
- waterndvithreshold 0.01
- waterreflectancevariationthreshold 0.04
- maximumsunglintreflectance 0.12
- srtmgaussiansigma 10
- srtmpercent 51
- watermasknumber 10
- minpercentpossiblewater 50
- waterslopethreshold 0.01
- waterredreflectancethreshold 0.05
- reall2nodata -10

6.5.60.4 Liste des produits de sortie

- was TVA/apTvWaterMaskingImageFilterTest-WAS.tif
- pwa TVA/apTvWaterMaskingImageFilterTest-PWA.tif
- twa TVA/apTvWaterMaskingImageFilterTest-TWA.tif

6.5.60.5 Prérequis

Il n'y a pas de prérequis.

6.5.60.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.60.7 Epsilon utilisé sur la non regression

0

6.5.60.8 Vérifications à effectuer

Le test génère en sortie le masque d'eau du produit.

6.5.60.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvWaterMask

6.5.60.10 Journal d'essai de la recette

6.5.60.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.60.10.2 Conclusion du déroulement du test

RAS

6.5.60.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.60.11 Exigences

Ce test couvre les exigences suivantes : ST-169-WaterMasking-0010 (C) ; ST-169-WaterMasking-0020 (C) ; ST-169-WaterMasking-0030(C) ; ST-169-WaterMasking-0040 (C) ; ST-169-WaterMasking-0050 (C) ; ST-169-WaterMasking-0060 (C) ; ST-169-WaterMasking-0060 (C) ;

6.5.61 apTvWaterVaporPostPro

6.5.61.1 Objectif

Validation de l'application "WaterVaporPostPro"

6.5.61.2 Description

Le module "WaterVaporPostPro" améliore le masque de vapeur d'eau

6.5.61.3 Liste des données d'entrées

- tocr WaterVaporPostProcessing/31TFJ_20150908_S2A_L1Csimu_surf_240m.tif
- vapin WaterVaporPostProcessing/31TFJ_20150908_S2A_L1Csimu_surf_VAP_SUB.tif
- cldsum WaterVaporPostProcessing/31TFJ_20150908_S2A_L1Csimu_toa_240m.CLDSum.tif
- defaultwatervapor 0
- threshold 200
- thresholdref 100
- referencebandcode 8
- bandcode 9
- nodata 0
- sevalidradius 1
- initwindowradius 2
- maxwindowradius 16

6.5.61.4 Liste des produits de sortie

- mask TVA/apTvWaterVaporPostPro_mask_240m.tif
- vap TVA/apTvWaterVaporPostPro_VAP_Filtered_240m.tif

6.5.61.5 Prérequis

Il n'y a pas de prérequis.

6.5.61.6 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

6.5.61.7 Epsilon utilisé sur la non regression

0

6.5.61.8 Vérifications à effectuer

Le test génère en sortie l'image de vapeur d'eau ainsi que le masque.

6.5.61.9 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R apTvWaterVaporPostPro

6.5.61.10 Journal d'essai de la recette

6.5.61.10.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

6.5.61.10.2 Conclusion du déroulement du test

RAS

6.5.61.10.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

6.5.61.11 Exigences

Ce test couvre les exigences suivantes :

LES ESSAIS DE VALIDATION

7.1 Objectifs

Les essais menés durant la phase d'intégration et de validation sont des essais dits fonctionnels et/ou numériques. Ils permettent de tester toutes les fonctionnalités du logiciel avec variation des paramètres d'entrée et des contextes d'utilisation. Ils couvrent les cas d'activation nominaux, les cas dégradés, les tests de performances. Ils permettent de répondre à la question « est-ce que le logiciel répond bien au besoin ? ».

L'objectif général de la validation est de garantir que le ou les composants implémentés fournissent des résultats exacts. Des tests de non-régression sont également mis en place afin de garantir l'exactitude des résultats au fur et à mesure des développements. Cette phase de validation permet notamment de :

- Vérifier l'adéquation des fonctionnalités offertes par le système aux exigences,
- Garantir, en complément des différents essais réalisés au niveau des composants, une vérification des exigences compréhensible par l'utilisateur,
- Valider les interfaces avec les entités externes,
- Pérenniser un ensemble de scripts/jeux de données/résultats pour chaque essai afin de pouvoir les rejouer ultérieurement en cas d'évolution du système,
- Apprêter le système pour son entrée en recette.

7.2 Description

Le processus de validation est itératif. Il est initialisé pendant la phase de conception, précisé tout au long de la progression du développement, et complété pour la livraison sur site. Le processus est composé des étapes suivantes :

- Définition des tests : cette activité consiste à prévoir les tests qui devront démontrer que le système est conforme aux exigences utilisateur. Cette définition est réalisée en deux étapes :
- La première consiste à produire un plan de validation. Elle s'attachera notamment à définir et identifier les essais unitaires et données constituant le plan de validation. Cette étape est réalisée durant les phases de conception et éventuellement complétée au cours des développements,
- La deuxième consiste à produire les plans d'essais détaillés (PED) de chacun des essais. Ces PED contiennent toutes les informations décrites section 3.6.1.1,
- Vérification de la couverture des essais : cette activité est réalisée avec l'outil Reqify, et une matrice de couverture « exigence/cas de tests » est produite.
- Les étapes d'exécution sont les suivantes :

- Génération de la version logicielle à valider par l'équipe projet à partir des sources gérés par l'équipe de gestion en configuration CS (la procédure de génération fait l'objet elle-même d'un test),
- Installation du logiciel (l'installation fait l'objet elle-même d'un test),
- Pour chaque essai de validation :
 - Installation des contextes d'essais, tels que définis dans le PED de l'essai,
 - Génération automatique du journal d'essai en début d'essai en fonction de sa description dans le PED,
 - Mise en œuvre de l'essai et rédaction de la partie résultats obtenus dans le journal d'essai,
 - Récupération des contextes de fin d'essai et exécution éventuelle de la procédure de vérification des résultats (comparaison du contexte récupéré par rapport à un contexte de référence).
 - Mise à jour du rapport de la campagne d'essais.

La phase de validation conduit à la mise en œuvre de l'ensemble des cas de tests identifiés. Ces cas de tests sont complétés et finalisés lors de cette phase. Les données et outils de validation sont mis en œuvre et adaptés le cas échéant. Les tests et données de validation sont maintenus pendant l'ensemble du cycle de vie du logiciel. Ce processus est tout d'abord mis en œuvre sur la machine de développement à CS sur laquelle est également réalisée la recette usine des chaînes MAJA L2, L3 ainsi que des outils associés. Suite à une phase d'intégration sur la configuration cible CNES, les tests associés à des produits simulés (choix du capteur) pourront être exécutés. La recette site sera réalisée sur la configuration cible CNES.

7.3 Formalisation des tests de validation

La nomenclature adoptée pour l'identification des essais d'intégration et de validation est la suivante :

S2-<traitement>-<num d'essai>-<nature>-<type>-<produit>-<thématique>_<MODE>

Ex : S2-L2INIT-001-F-N-VENUS-CLOUD

Avec :

Traitement identifie le type de traitement et peut prendre les valeurs :

- L2INIT : Chaîne L2 en mode Init,
- L2NOMINAL : Chaîne L2 en mode Nominal,
- L2BACKWARD : Chaîne L2 en mode Backward,

Numéro d'essai :

- 000 : valeur codée sur 3 caractères, valeur unique pour chaque <traitement>

Nature identifie la nature du test et peut prendre les valeurs :

- F : test fonctionnel,
- N : test numérique,

Type identifie le type de l'essai et peut prendre les valeurs :

- N : nominal,
- D : dégradé,
- L : aux limites

Produit identifie le type de produit image utilisé et peut prendre les valeurs :

- VENUS : produit image Venus,

- FORMOSAT : produit image Formosat-2,
- LANDSAT : produit image Landsat,
- SENTINEL2 : produit image Sentinel2

Thématique est une abréviation du thème autour duquel s'organise le test, sur 10 caractères maximum.

Le mode peut prendre les valeurs :

- CHAIN : exécution de la chaîne
- COMP_ASCII : non regression des fichiers textes
- COMP_IMAGE : Comparaison des images

7.4 LISTE DES ESSAIS

Ce chapitre présente la liste des essais mis en place pour valider MAJA.

num	Val	Id	Commentaire
1	OK	pyTv-S2-L2NOMINAL-001-SENTINEL2-ALGO_CHAIN	Validation fonctionnelle et numérique de la chaîne L2 avec un produit SENTINEL2 natif PDGS en mode NOMINAL.
2	OK	pyTv-S2-L2NOMINAL-001-SENTINEL2-ALGO_COMP_IMAGE	Non regression Images du test CHAIN correspondant
3	OK	pyTv-S2-L2NOMINAL-001-SENTINEL2-ALGO_COMP_ASCII	Non regression Texte du test CHAIN correspondant
4	OK	pyTv-S2-L2INIT-011-SENTINEL2-MUSCATE-GENERAL_CHAIN	Validation fonctionnelle et numérique avec un produit SENTINEL2 Muscate en mode INIT.
5	OK	pyTv-S2-L2INIT-011-SENTINEL2-MUSCATE-GENERAL_COMP_IMAGE	Non regression Images du test CHAIN correspondant
6	OK	pyTv-S2-L2INIT-011-SENTINEL2-MUSCATE-GENERAL_COMP_ASCII	Non regression Texte du test CHAIN correspondant
7	OK	pyTv-S2-L2INIT-012-SENTINEL2-MUSCATE-CAMS_CHAIN	Validation fonctionnelle et numérique avec un produit SENTINEL2 Muscate en mode INIT avec CAMS.
8	OK	pyTv-S2-L2INIT-012-SENTINEL2-MUSCATE-CAMS_COMP_IMAGE	Non regression Images du test CHAIN correspondant
9	OK	pyTv-S2-L2INIT-012-SENTINEL2-MUSCATE-CAMS_COMP_ASCII	Non regression Texte du test CHAIN correspondant
10	OK	pyTv-S2-L2INIT-012-SENTINEL2-MUSCATE-NEWCAMS_CHAIN	Validation fonctionnelle et numérique avec un produit SENTINEL2 Muscate en mode INIT avec NewCAMS.
11	OK	pyTv-S2-L2INIT-012-SENTINEL2-MUSCATE-NEWCAMS_COMP_IMAGE	Non regression Images du test CHAIN correspondant
12	OK	pyTv-S2-L2INIT-012-SENTINEL2-MUSCATE-NEWCAMS_COMP_ASCII	Non regression Texte du test CHAIN correspondant
13	OK	pyTv-S2-L2NOMINAL-011-F-N-SENTINEL2_MUSCATE-GENERAL_CHAIN	Validation fonctionnelle et numérique avec un produit SENTINEL2 Muscate en mode NOMINAL.
14	OK	pyTv-S2-L2NOMINAL-011-F-N-SENTINEL2_MUSCATE-GENERAL_COMP_IMAGE	Non regression Images du test CHAIN correspondant
15	OK	pyTv-S2-L2NOMINAL-011-F-N-SENTINEL2_MUSCATE-GENERAL_COMP_ASCII	Non regression Texte du test CHAIN correspondant
16	OK	pyTv-S2-L2BACKWARD-011-F-N-SENTINEL2_MUSCATE-GENERAL_CHAIN	Validation fonctionnelle et numérique avec un produit SENTINEL2 Muscate en mode BACKWARD.
17	OK	pyTv-S2-L2BACKWARD-011-F-N-SENTINEL2_MUSCATE-GENERAL_COMP_IMAGE	Non regression Images du test CHAIN correspondant
18	OK	pyTv-S2-L2BACKWARD-011-F-N-SENTINEL2_MUSCATE-GENERAL_COMP_ASCII	Non regression Texte du test CHAIN correspondant

num	Val	Id	Commentaire
19	OK	pyTv-S2-L2NOMINAL-011-F-N-SENTINEL2_MUSCATE-MIXRES_CHAIN	Validation fonctionnelle et numérique avec un produit SENTINEL2 Muscate en mode NOMINAL.
20	OK	pyTv-S2-L2NOMINAL-011-F-N-SENTINEL2_MUSCATE-MIXRES_COMP_IMAGE	Non regression Images du test CHAIN correspondant
21	OK	pyTv-S2-L2NOMINAL-011-F-N-SENTINEL2_MUSCATE-MIXRES_COMP_ASCII	Non regression Texte du test CHAIN correspondant
22	OK	pyTv-S2-L2NOMINAL-011-F-N-SENTINEL2_MUSCATE-LUTDIRCOR_CHAIN	Validation fonctionnelle et numérique avec un produit SENTINEL2 Muscate en mode NOMINAL en utilisant les LUTs pour la correction directionnelle.
23	OK	pyTv-S2-L2NOMINAL-011-F-N-SENTINEL2_MUSCATE-LUTDIRCOR_COMP_IMAGE	Non regression Images du test CHAIN correspondant
24	OK	pyTv-S2-L2NOMINAL-011-F-N-SENTINEL2_MUSCATE-LUTDIRCOR_COMP_ASCII	Non regression Texte du test CHAIN correspondant
25	OK	pyTv-S2-L2INIT-CROSSWRITING-SENTINEL2-SENTINEL2MUSCATE	Validation fonctionnelle et numérique avec un produit SENTINEL2 Natif vers Muscate en mode INIT .
26	OK	pyTv-S2-L2INIT-CROSSWRITING-LANDSAT8-LANDSAT8MUSCATE	Validation fonctionnelle et numérique avec un produit LANDSAT8 Natif vers Muscate en mode INIT .
27	OK	pyTv-S2-L2INIT-CROSSWRITING-VENUS-VENUSMUSCATE	Validation fonctionnelle et numérique avec un produit VENUS Natif vers Muscate en mode INIT.
28	OK	pyTv-VE-L2INIT-VENUS-MUSCATE_CHAIN	Validation fonctionnelle et numérique avec un produit VENUS Muscate en mode INIT.
29	OK	pyTv-VE-L2INIT-VENUS-MUSCATE_COMP_IMAGE	Non regression Images du test CHAIN correspondant
30	OK	pyTv-VE-L2INIT-VENUS-MUSCATE_COMP_ASCII	Non regression Texte du test CHAIN correspondant
31	OK	pyTv-STARTMAJA_CHAIN	Validaton fonctionnelle et numérique de la chaîne StartMaja avec un produit Sentinel en mode INIT et NOMINAL
32	OK	pyTv-STARTMAJA_COMP_IMAGE	Non regression Images du test CHAIN correspondant

7.4.1 pyTv-S2-L2NOMINAL-001-SENTINEL2-ALGO_CHAIN

7.4.1.1 Titre

Validation fonctionnelle et numérique de la chaîne L2 (Pre-PostProcessing, entrée/sorties) avec un produit SENTINEL2 natif PDGS en mode NOMINAL. Les paramètres de la couche Standalone sont passés en ligne de commande (pas de JobOrder).

7.4.1.2 Objectif

L'objectif de cet essai est de valider le fonctionnement global de la chaîne L2 en mode NOMINAL sur des produits SENTINEL2 natifs PDGS comportant différentes résolutions. Plus particulièrement dans le cadre du développement d'un plug-in SENTINEL2 natif PDGS, ce test permet de vérifier que la chaîne est capable de relire un produits L2 au format déjà développé pour le plug-in SENTINEL2 mais généré par ce plug-in. Les produits L1 fournis en entrée sont conformes au document S2-PDGS-TAS-DI-PSD (12) (DA15). Les produits L2 fournis en entrée et générés en sortie sont conforme au format du plug-in Sentinel-2.

7.4.1.3 Description

Les options de traitement sont :

- Méthode SPECTROTEMPORAL pour l'estimation des aérosols,
- Correction de l'environnement et des pentes,
- Ecriture à la résolution L2.

7.4.1.4 Liste des données d'entrées

Produits :

- S2A_OPER_PRD_MSIL1C_PDMC_20161109T171237_R135_V20160924T074932_20160924T081448.SAFE
- S2A_OPER_SSC_L2VALD_36JTT____20160914.HDR
- S2__TEST_AUX_REFDE2_36JTT____5001.DBL
- S2__TEST_AUX_REFDE2_36JTT____5001.DBL.DIR

Gipps:

- S2A_TEST_GIP_L2DIFT_L_CONTINEN_50003_20150703_21000101.DBL
- S2A_TEST_GIP_L2WATV_L_CONTINEN_50003_20150703_21000101.DBL.DIR
- S2A_OPER_SSC_L2VALD_36JTT____20160914.DBL
- S2A_TEST_GIP_L2DIFT_L_CONTINEN_50003_20150703_21000101.DBL.DIR
- S2A_TEST_GIP_L2WATV_L_CONTINEN_50003_20150703_21000101.HDR
- S2A_OPER_SSC_L2VALD_36JTT____20160914.DBL.DIR
- S2A_TEST_GIP_L2DIFT_L_CONTINEN_50003_20150703_21000101.HDR
- S2A_TEST_GIP_L2DIRT_L_CONTINEN_50003_20150703_21000101.DBL
- S2A_TEST_GIP_CKEXTL_S_36JTT____50001_00000000_99999999.EEF
- S2A_TEST_GIP_L2DIRT_L_CONTINEN_50003_20150703_21000101.DBL.DIR
- S2A_TEST_GIP_CKQLTL_S_36JTT____50001_00000000_99999999.EEF
- S2A_TEST_GIP_L2DIRT_L_CONTINEN_50003_20150703_21000101.HDR
- S2__TEST_EXO_METDTA_20090101T000000_20111231T230000.DBL
- S2A_TEST_GIP_L2ALBD_L_CONTINEN_50004_20150703_21000101.DBL
- S2A_TEST_GIP_L2SMAC_L_ALLSITES_50003_20150703_21000101.EEF
- S2__TEST_EXO_METDTA_20090101T000000_20111231T230000.DBL.BIN

- S2A_TEST_GIP_L2ALBD_L_CONTINEN_50004_20150703_21000101.DBL.DIR
- S2A_TEST_GIP_L2TOCR_L_CONTINEN_50003_20150703_21000101.DBL
- S2__TEST_EXO_METDTA_20090101T000000_20111231T230000.HDR
- S2A_TEST_GIP_L2ALBD_L_CONTINEN_50004_20150703_21000101.HDR
- S2A_TEST_GIP_L2TOCR_L_CONTINEN_50003_20150703_21000101.DBL.DIR
- S2__TEST_GIP_L2SITE_S_36JTT____50001_00000000_99999999.EEF
- S2A_TEST_GIP_L2TOCR_L_CONTINEN_50003_20150703_21000101.HDR
- S2A_TEST_GIP_L2COMM_L_ALLSITES_50008_20150703_21000101.EEF
- S2A_TEST_GIP_L2WATV_L_CONTINEN_50003_20150703_21000101.DBL

7.4.1.5 Liste des produits de sortie

Produit L2A SENTINEL2 Natif

7.4.1.6 Prérequis

Il n'y a pas de prérequis.

7.4.1.7 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

7.4.1.8 Epsilon utilisé sur la non regression

0.0001

7.4.1.9 Vérifications à effectuer

Les tests COMP_ASCII et COMP_IMAGE associés permettent de valider la non regression.

7.4.1.10 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R pyTv-S2-L2NOMINAL-001-SENTINEL2-ALGO_CHAIN

7.4.1.11 Exigences

Ce test couvre les exigences suivantes : MACCS-Exigence 50 (C) ; MACCS-Exigence 550 (C) ; MACCS-Exigence 630 (C) ;

7.4.1.12 Journal d'essai de la recette

7.4.1.12.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

7.4.1.12.2 Conclusion du déroulement du test

RAS

7.4.1.12.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

7.4.2 pyTv-S2-L2INIT-011-SENTINEL2-MUSCATE-GENERAL_CHAIN

7.4.2.1 Titre

Validation fonctionnelle et numérique de la chaîne L2 (Pre-PostProcessing, entrée/sorties) avec un produit L1C SENTINEL2 S2A au format Muscate en mode INIT.

7.4.2.2 Objectif

L'objectif de cet essai est de valider le fonctionnement global de la chaîne L2 avec un produit SENTINEL2 au format Muscate.

7.4.2.3 Description

Les options de traitement sont :

- Méthode SPECTROTEMPORAL pour l'estimation des aérosols,
- Correction de l'environnement et des pentes,
- Ecriture aux différentes résolutions L2,
- activation de la correction des cirrus
- Correction Bidirectionnelle sur la minilut

7.4.2.4 Liste des données d'entrées

Les images utilisées en entrée sont les suivantes :

- SENTINEL2B_20171008-105012-463_L1C_T31TCH_C_V1-0/SENTINEL2B_20171008105012-463_L1C_T31TCH_C_V1-0_MTD_ALL.xml
- S2__TEST_AUX_REFDE2_T31TCH_0001.DBL

Les fichiers GIPPs utilisés en entrée sont les suivants :

- S2__TEST_GIP_L2SITE_S_31TJF____10001_00000000_99999999.EEF
- S2B_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.DBL
- S2B_TEST_GIP_L2COMM_L_ALLSITES_10008_20150703_21000101.EEF
- S2B_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2SMAC_L_ALLSITES_10001_20150703_21000101.EEF
- S2B_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_CKEXTL_S_31TJF____10001_20150703_21000101.EEF
- S2B_TEST_GIP_CKQLTL_S_31TJF____10005_20150703_21000101.EEF

7.4.2.5 Liste des produits de sortie

Produit SENTINEL2 au format Muscate

7.4.2.6 Prérequis

Il n'y a pas de prérequis.

7.4.2.7 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

7.4.2.8 Epsilon utilisé sur la non regression

0.0001

7.4.2.9 Vérifications à effectuer

Les tests COMP_ASCII et COMP_IMAGE associés permettent de valider la non regression

7.4.2.10 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : `ctest -R pyTv-S2-L2INIT-011-SENTINEL2-MUSCATE-GENERAL_CHAIN`

7.4.2.11 Exigences

Ce test couvre les exigences suivantes : MACCS-Exigence 640 (C) ; MACCS-Exigence 650 (C) ; MACCS-Exigence 50 (C) ; MACCS-Exigence 630 (C) ;

7.4.2.12 Journal d'essai de la recette

7.4.2.12.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

7.4.2.12.2 Conclusion du déroulement du test

RAS

7.4.2.12.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

7.4.3 pyTv-S2-L2INIT-012-SENTINEL2-MUSCATE-CAMS_CHAIN

7.4.3.1 Titre

Validation fonctionnelle et numérique de la chaîne L2 (Pre-PostProcessing, entrée/sorties) avec un produit L1C SENTINEL2 S2A au format Muscate en mode INIT en utilisant des données CAMS.

7.4.3.2 Objectif

L'objectif de cet essai est de valider le fonctionnement global de la chaîne L2 avec un produit SENTINEL2 au format Muscate.

7.4.3.3 Description

Les options de traitement sont :

- Méthode SPECTROTEMPORAL pour l'estimation des aérosols,
- Correction de l'environnement et des pentes,
- Ecriture aux différentes résolutions L2,
- activation de la correction des cirrus
- Correction Bidirectionnelle sur la minilut

7.4.3.4 Liste des données d'entrées

- S2B_TEST_GIP_CKEXTL_S_31TJF____10001_20150703_21000101.EEF
- S2B_TEST_GIP_L2DIFT_L_BLACKCAR_50002_00000000_99999999.HDR
- S2B_TEST_GIP_L2DIRT_L_SULPHATE_50002_00000000_99999999.DBL.DIR
- S2B_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_CKQLTL_S_31TJF____10005_20150703_21000101.EEF
- S2B_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2DIRT_L_SULPHATE_50002_00000000_99999999.HDR
- S2__TEST_AUX_REFDE2_T31TCH_0001.DBL
- S2B_TEST_GIP_L2ALBD_L_BLACKCAR_50002_00000000_99999999.DBL
- S2B_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.DBL.DIR
- S2B_TEST_GIP_L2SMAC_L_ALLSITES_10001_20150703_21000101.EEF
- S2__TEST_AUX_REFDE2_T31TCH_0001.DBL.DIR
- S2B_TEST_GIP_L2ALBD_L_BLACKCAR_50002_00000000_99999999.DBL.DIR
- S2B_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_L2TOCR_L_BLACKCAR_50002_00000000_99999999.DBL
- S2__TEST_AUX_REFDE2_T31TCH_0001.HDR
- S2B_TEST_GIP_L2ALBD_L_BLACKCAR_50002_00000000_99999999.HDR
- S2B_TEST_GIP_L2DIFT_L_SULPHATE_50002_00000000_99999999.DBL
- S2B_TEST_GIP_L2TOCR_L_BLACKCAR_50002_00000000_99999999.DBL.DIR
- S2__TEST_EXO_CAMS_20171007T000000_20171007T235959.DBL
- S2B_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.DBL
- S2B_TEST_GIP_L2DIFT_L_SULPHATE_50002_00000000_99999999.DBL.DIR
- S2B_TEST_GIP_L2TOCR_L_BLACKCAR_50002_00000000_99999999.HDR
- S2__TEST_EXO_CAMS_20171007T000000_20171007T235959.DBL.DIR
- S2B_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.DBL.DIR
- S2B_TEST_GIP_L2DIFT_L_SULPHATE_50002_00000000_99999999.HDR
- S2B_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.DBL
- S2__TEST_EXO_CAMS_20171007T000000_20171007T235959.HDR
- S2B_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.HDR
- S2B_TEST_GIP_L2DIRT_L_BLACKCAR_50002_00000000_99999999.DBL
- S2B_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.DBL.DIR
- S2__TEST_EXO_CAMS_20171008T000000_20171008T235959.DBL
- S2B_TEST_GIP_L2ALBD_L_SULPHATE_50002_00000000_99999999.DBL
- S2B_TEST_GIP_L2DIRT_L_BLACKCAR_50002_00000000_99999999.DBL.DIR
- S2B_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.HDR

- S2__TEST_EXO_CAMS_20171008T000000_20171008T235959.DBL.DIR
- S2B_TEST_GIP_L2ALBD_L_SULPHATE_50002_00000000_99999999.DBL.DIR
- S2B_TEST_GIP_L2DIRT_L_BLACKCAR_50002_00000000_99999999.HDR
- S2B_TEST_GIP_L2TOCR_L_SULPHATE_50002_00000000_99999999.DBL
- S2__TEST_EXO_CAMS_20171008T000000_20171008T235959.HDR
- S2B_TEST_GIP_L2ALBD_L_SULPHATE_50002_00000000_99999999.HDR
- S2B_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2TOCR_L_SULPHATE_50002_00000000_99999999.DBL.DIR
- S2__TEST_GIP_L2SITE_S_31TJF____10001_00000000_99999999.EEF
- S2B_TEST_GIP_L2COMM_L_ALLSITES_10008_20150703_21000101.EEF
- S2B_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.DBL.DIR
- S2B_TEST_GIP_L2TOCR_L_SULPHATE_50002_00000000_99999999.HDR
- SENTINEL2B_20171008-105012-463_L1C_T31TCH_C_V1-0
- S2B_TEST_GIP_L2DIFT_L_BLACKCAR_50002_00000000_99999999.DBL
- S2B_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2DIFT_L_BLACKCAR_50002_00000000_99999999.DBL.DIR
- S2B_TEST_GIP_L2DIRT_L_SULPHATE_50002_00000000_99999999.DBL
- S2B_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.DBL.DIR

7.4.3.5 Liste des produits de sortie

Produit L2A SENTINEL Muscate

7.4.3.6 Prérequis

Il n'y a pas de prérequis.

7.4.3.7 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

7.4.3.8 Epsilon utilisé sur la non regression

0.0001

7.4.3.9 Vérifications à effectuer

Les tests COMP_ASCII et COMP_IMAGE associés permettent de valider la non regression

7.4.3.10 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R pyTv-S2-L2INIT-012-SENTINEL2-MUSCATE-CAMS_CHAIN

7.4.3.11 Exigences

Ce test couvre les exigences suivantes : Néant

7.4.3.12 Journal d'essai de la recette

7.4.3.12.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

7.4.3.12.2 Conclusion du déroulement du test

RAS

7.4.3.12.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

7.4.4 pyTv-S2-L2INIT-012-SENTINEL2-MUSCATE-CAMS_CHAIN

7.4.4.1 Titre

Validation fonctionnelle et numérique de la chaîne L2 (Pre-PostProcessing, entrée/sorties) avec un produit L1C SENTINEL2 S2A au format Muscate en mode INIT en utilisant des données CAMS au nouveau format.

7.4.4.2 Objectif

L'objectif de cet essai est de valider le fonctionnement global de la chaîne L2 avec un produit SENTINEL2 au format Muscate.

7.4.4.3 Description

Les options de traitement sont :

- Méthode SPECTROTEMPORAL pour l'estimation des aérosols,
- Correction de l'environnement et des pentes,
- Ecriture aux différentes résolutions L2,
- activation de la correction des cirrus
- Correction Bidirectionnelle sur la minilut

7.4.4.4 Liste des données d'entrées

- S2B_TEST_GIP_CKEXTL_S_31TJF____10001_20150703_21000101.EEF
- S2B_TEST_GIP_L2DIFT_L_BLACKCAR_50002_00000000_99999999.HDR
- S2B_TEST_GIP_L2DIRT_L_SULPHATE_50002_00000000_99999999.DBL.DIR
- S2B_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_CKQLTL_S_31TJF____10005_20150703_21000101.EEF
- S2B_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2DIRT_L_SULPHATE_50002_00000000_99999999.HDR
- S2__TEST_AUX_REFDE2_T31TCH_0001.DBL
- S2B_TEST_GIP_L2ALBD_L_BLACKCAR_50002_00000000_99999999.DBL
- S2B_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.DBL.DIR
- S2B_TEST_GIP_L2SMAC_L_ALLSITES_10001_20150703_21000101.EEF
- S2__TEST_AUX_REFDE2_T31TCH_0001.DBL.DIR
- S2B_TEST_GIP_L2ALBD_L_BLACKCAR_50002_00000000_99999999.DBL.DIR
- S2B_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_L2TOCR_L_BLACKCAR_50002_00000000_99999999.DBL
- S2__TEST_AUX_REFDE2_T31TCH_0001.HDR
- S2B_TEST_GIP_L2ALBD_L_BLACKCAR_50002_00000000_99999999.HDR
- S2B_TEST_GIP_L2DIFT_L_SULPHATE_50002_00000000_99999999.DBL
- S2B_TEST_GIP_L2TOCR_L_BLACKCAR_50002_00000000_99999999.DBL.DIR
- S2__TEST_EXO_CAMS_20171007T000000_20171007T235959.DBL
- S2B_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.DBL
- S2B_TEST_GIP_L2DIFT_L_SULPHATE_50002_00000000_99999999.DBL.DIR
- S2B_TEST_GIP_L2TOCR_L_BLACKCAR_50002_00000000_99999999.HDR
- S2__TEST_EXO_CAMS_20171007T000000_20171007T235959.DBL.DIR
- S2B_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.DBL.DIR
- S2B_TEST_GIP_L2DIFT_L_SULPHATE_50002_00000000_99999999.HDR
- S2B_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.DBL

- S2__TEST_EXO_CAMS_20171007T000000_20171007T235959.HDR
- S2B_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.HDR
- S2B_TEST_GIP_L2DIRT_L_BLACKCAR_50002_00000000_99999999.DBL
- S2B_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.DBL.DIR
- S2__TEST_EXO_CAMS_20171008T000000_20171008T235959.DBL
- S2B_TEST_GIP_L2ALBD_L_SULPHATE_50002_00000000_99999999.DBL
- S2B_TEST_GIP_L2DIRT_L_BLACKCAR_50002_00000000_99999999.DBL.DIR
- S2B_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.HDR
- S2__TEST_EXO_CAMS_20171008T000000_20171008T235959.DBL.DIR
- S2B_TEST_GIP_L2ALBD_L_SULPHATE_50002_00000000_99999999.DBL.DIR
- S2B_TEST_GIP_L2DIRT_L_BLACKCAR_50002_00000000_99999999.HDR
- S2B_TEST_GIP_L2TOCR_L_SULPHATE_50002_00000000_99999999.DBL
- S2__TEST_EXO_CAMS_20171008T000000_20171008T235959.HDR
- S2B_TEST_GIP_L2ALBD_L_SULPHATE_50002_00000000_99999999.HDR
- S2B_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2TOCR_L_SULPHATE_50002_00000000_99999999.DBL.DIR
- S2__TEST_GIP_L2SITE_S_31TJF____10001_00000000_99999999.EEF
- S2B_TEST_GIP_L2COMM_L_ALLSITES_10008_20150703_21000101.EEF
- S2B_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.DBL.DIR
- S2B_TEST_GIP_L2TOCR_L_SULPHATE_50002_00000000_99999999.HDR
- SENTINEL2B_20171008-105012-463_L1C_T31TCH_C_V1-0
- S2B_TEST_GIP_L2DIFT_L_BLACKCAR_50002_00000000_99999999.DBL
- S2B_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2DIFT_L_BLACKCAR_50002_00000000_99999999.DBL.DIR
- S2B_TEST_GIP_L2DIRT_L_SULPHATE_50002_00000000_99999999.DBL
- S2B_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.DBL.DIR

7.4.4.5 Liste des produits de sortie

Produit L2A SENTINEL Muscate

7.4.4.6 Prérequis

Il n'y a pas de prérequis.

7.4.4.7 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

7.4.4.8 Epsilon utilisé sur la non regression

0.0001

7.4.4.9 Vérifications à effectuer

Les tests COMP_ASCII et COMP_IMAGE associés permettent de valider la non regression

7.4.4.10 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : `ctest -R pyTv-S2-L2INIT-012-SENTINEL2-MUSCATE-NEWCAMS_CHAIN`

7.4.4.11 Exigences

Ce test couvre les exigences suivantes : Néant

7.4.4.12 Journal d'essai de la recette

7.4.4.12.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

7.4.4.12.2 Conclusion du déroulement du test

RAS

7.4.4.12.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

7.4.5 pyTv-S2-L2NOMINAL-011-F-N-SENTINEL2_MUSCATE-GENERAL_CHAIN

7.4.5.1 Titre

Validation fonctionnelle et numérique de la chaîne L2 (Pre-PostProcessing, entrée/sorties) avec un produit SENTINEL2 Muscate en mode NOMINAL.

7.4.5.2 Objectif

L'objectif de cet essai est de valider le fonctionnement global de la chaîne L2 en mode NOMINAL sur des produits SENTINEL2 Muscate comportant différentes résolutions. Plus particulièrement dans le cadre du développement d'un plug-in SENTINEL2 Muscate, ce test permet de vérifier que la chaîne est capable de relire un produits L2 au format déjà développé pour le plug-in SENTINEL2 mais généré par ce plug-in.

7.4.5.3 Description

Les options de traitement sont :

- Méthode SPECTROTEMPORAL pour l'estimation des aérosols,
- Correction de l'environnement et des pentes,
- Ecriture à la résolution L2,
- activation de la correction des cirrus

7.4.5.4 Liste des données d'entrées

Les images utilisées en entrée sont les suivantes :

- SENTINEL2B_20171010-105012-463_L1C_T31TCH_C_V1-0/SENTINEL2B_20171010105012-463_L1C_T31TCH_C_V1-0_MTD_ALL.xml
- SENTINEL2B_20171008-105012-463_L2A_T31TCH_C_V1-0/SENTINEL2B_20171008105012-463_L2A_T31TCH_C_V1-0_MTD_ALL.xml
- S2__TEST_AUX_REFDE2_T31TCH_0001.DBL

Les fichiers GIPPs utilisés en entrée sont les suivants :

- S2A_TEST_GIP_CKEXTL_S_31TJF____10001_20150703_21000101.EEF
- S2A_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.DBL
- S2A_TEST_GIP_CKQLTL_S_31TJF____10005_20150703_21000101.EEF
- S2A_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.DBL.DIR
- S2B_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.DBL.DIR
- S2A_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.DBL
- S2A_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.HDR

- S2B_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.HDR
- S2A_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.HDR
- S2A_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.DBL
- S2A_TEST_GIP_L2COMM_L_ALLSITES_10008_20150703_21000101.EEF
- S2B_TEST_GIP_CKEXTL_S_31TJF____10001_20150703_21000101.EEF
- S2B_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.DBL.DIR
- S2A_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_CKQLTL_S_31TJF____10005_20150703_21000101.EEF
- S2B_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.HDR
- S2A_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.DBL
- S2B_TEST_GIP_L2SMAC_L_ALLSITES_10001_20150703_21000101.EEF
- S2__TEST_GIP_L2SITE_S_31TJF____10001_00000000_99999999.EEF
- S2A_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.DBL.DIR
- S2B_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.DBL
- S2A_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.HDR
- S2B_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.DBL.DIR
- S2A_TEST_GIP_L2SMAC_L_ALLSITES_10001_20150703_21000101.EEF
- S2B_TEST_GIP_L2COMM_L_ALLSITES_10008_20150703_21000101.EEF
- S2B_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.HDR

7.4.5.5 Liste des produits de sortie

Produit SENTINEL2 L2A MUSCATE

7.4.5.6 Prérequis

Il n'y a pas de prérequis.

7.4.5.7 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

7.4.5.8 Epsilon utilisé sur la non regression

0.0001

7.4.5.9 Vérifications à effectuer

Le test génère en sortie une image contenant les angles.

7.4.5.10 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R pyTv-S2-L2NOMINAL-011-F-N-SENTINEL2_MUSCATE-GENERAL_CHAIN

7.4.5.11 Exigences

Ce test couvre les exigences suivantes : MACCS-Exigence 640 (C) ; MACCS-Exigence 650 (C) ;

7.4.5.12 Journal d'essai de la recette**7.4.5.12.1 Notes sur le déroulement du test**

Rien de particulier n'a été noté lors du déroulement du test.

7.4.5.12.2 Conclusion du déroulement du test

RAS

7.4.5.12.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

7.4.6 pyTv-S2-L2BACKWARD-011-F-N-SENTINEL2_MUSCATE-GENERAL_CHAIN**7.4.6.1 Titre**

Validation fonctionnelle et numérique de la chaîne L2 (Pre-PostProcessing, entrée/sorties) avec des produits SENTINEL2 Muscate en mode Backward, avec AOT en mode SPECTROTEMPORAL.

7.4.6.2 Objectif

L'objectif de cet essai est de valider le fonctionnement global de la chaîne L2 avec des produits SENTINEL2 au format Muscate. Cet essai enchaîne les traitements de Pre/Post processing et l'exécution de la chaîne L2 en mode BACKWARD (L2BACK). Le mode BACKWARD doit lancer une première fois la chaîne en mode INIT sur l'image la plus récente au format PDGS conforme à DA15 puis basculer en mode NOMINAL pour traiter toutes les images précédentes jusqu'à traiter la première image et généré le produit L2 conformément au format déjà développé pour les produits L2 des plug-ins Muscate. Ce test permet donc de contrôler l'enchaînement des différents modes. Il permet aussi de contrôler que toutes les images intermédiaires (dates D-N à D-1) sont bien produites à la résolution réduite uniquement et que la chaîne est capable de relire les produits générés par ce nouveau plug-in SENTINEL2.

7.4.6.3 Description

Les options de traitement sont :

- Méthode d'estimation des aérosols : SPECTROTEMPORAL
- Pas d'extension du masque des nuages,
- Lancement des traitements de correction de l'environnement et des pentes,
- Ecriture à la résolution L2.

7.4.6.4 Liste des données d'entrées

Les images utilisées en entrée sont les suivantes :

- SENTINEL2B_20171008-105012-463_L1C_T31TCH_C_V1-0/SENTINEL2B_20171008105012-463_L1C_T31TCH_C_V1-0_MTD_ALL.xml
- SENTINEL2B_20171010-105012-463_L1C_T31TCH_C_V1-0/SENTINEL2B_20171010105012-463_L1C_T31TCH_C_V1-0_MTD_ALL.xml
- S2__TEST_AUX_REFDE2_T31TCH_0001.DBL

Les fichiers GIPPs utilisés en entrée sont les suivants :

- S2__TEST_GIP_L2SITE_S_31TJF____10001_00000000_99999999.EEF
- S2A_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.DBL
- S2B_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.DBL
- S2A_TEST_GIP_L2COMM_L_ALLSITES_10008_20150703_21000101.EEF
- S2B_TEST_GIP_L2COMM_L_ALLSITES_10008_20150703_21000101.EEF
- S2A_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.DBL
- S2A_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.DBL
- S2A_TEST_GIP_L2SMAC_L_ALLSITES_10001_20150703_21000101.EEF
- S2B_TEST_GIP_L2SMAC_L_ALLSITES_10001_20150703_21000101.EEF
- S2A_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.DBL

- S2A_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.DBL
- S2A_TEST_GIP_CKEXTL_S_31TJF____10001_20150703_21000101.EEF
- S2B_TEST_GIP_CKEXTL_S_31TJF____10001_20150703_21000101.EEF
- S2A_TEST_GIP_CKQLTL_S_31TJF____10005_20150703_21000101.EEF
- S2B_TEST_GIP_CKQLTL_S_31TJF____10005_20150703_21000101.EEF

7.4.6.5 Liste des produits de sortie

Produit L2A au format SENTINEL2 Muscate

7.4.6.6 Prérequis

Il n'y a pas de prérequis.

7.4.6.7 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

7.4.6.8 Epsilon utilisé sur la non regression

0.0001

7.4.6.9 Vérifications à effectuer

Les tests COMP_ASCII et COMP_IMAGE associés permettent de valider la non regression

7.4.6.10 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : `cstest -R pyTv-S2-L2BACKWARD-011-F-N-SENTINEL2_MUSCATE-GENERAL_CHAIN`

7.4.6.11 Exigences

Ce test couvre les exigences suivantes :

MACCS-Exigence 640 (C) ; MACCS-Exigence 650 (C) ;MACCS-Exigence 30 (C) ; MACCS-Exigence 50 (C) ; MACCS-Exigence 70 (C) ; ST-169-Level1-0010 (C) ; ST-169-Level1-0030 (C) ; ST-169-Mode-0010 (C) ; ST-169-L1toL2-0010 (C) ;ST-169-DTM-0010 (C) ; ST-169-DTM-0020 (C) ; ST-169-DTM-0030 (C) ; ST-169-DTM-0040(C) ; ST-169-DTM-0050 (C) ; ST-169-Subsampling-0010 (C) ; ST-169-Subsampling-0020 (C) ;ST-169-Subsampling-0030 (C) ; ST-169-Subsampling-0040 (C) ; ST-169-Subsampling-0050 (C) ;ST-169-Subsampling-0060 (C) ; ST-169-AbsorptionCorr-0010 (C) ; ST-169-AbsorptionCorr-0030(C) ; ST-169-AbsorptionCorr-0040 (C) ; ST-169-AbsorptionCorr-0050 (C) ; ST-169-RayleighCorr-0010 (C) ; ST-169-GeometricFlags-0010 (C) ; ST-169-GeometricFlags-0020 (C) ; ST-169-GeometricFlags-0030 (C) ; ST-169-CirrusMaskS2-0010 (C) ; ST-169-CloudMasking-0010 (C) ;ST-169-CloudMasking-0100 (C) ; ST-169-CloudMasking-0200 (C) ; ST-169-CloudMasking-0500(C) ; ST-169-CloudMasking-0600 (C) ; ST-169-CloudMasking-0700 (C) ; ST-169-CloudMasking-1000 (C) ; ST-169-CloudMasking-1600 (C) ; ST-169-CloudMasking-1700 (C) ;

ST-169-CloudMasking-0300 (C) ; ST-169-CloudMasking-0400 (C) ; ST-169-CloudMasking-0410 (C) ; ST-169-CloudMasking-0420 (C) ; ST-169-CloudMasking-0430 (C) ; ST-169-CloudMasking-1100 (C) ; ST-169-CloudMasking-1110 (C) ; ST-169-CloudMasking-1120 (C) ; ST-169-CloudMasking-1130(C) ; ST-169-CloudMasking-1140 (C) ; ST-169-CloudMasking-1150 (C) ; ST-169-CloudMasking-1160 (C) ; ST-169-CloudMasking-1170 (C) ; ST-169-CloudMasking-1180 (C) ; ST-169-CloudMasking-1200 (C) ; ST-169-CloudMasking-1300 (C) ; ST-169-CloudMasking-1400 (C) ; ST-169-CloudMasking-1500 (C) ; ST-169-CloudMasking-1510 (C) ; ST-169-CloudMasking-1520 (C) ; ST-169-WaterMasking-0010 (C) ; ST-169-WaterMasking-0020 (C) ; ST-169-WaterMasking-0030(C) ; ST-169-WaterMasking-0040 (C) ; ST-169-WaterMasking-0050 (C) ; ST-169-WaterMasking-0060 (C) ; ST-169-RainDetection-0010 (C) ; ST-169-AOT-0010 (C) ; ST-169-AOT-0100 (C) ; ST-169-AOT-0200 (C) ; ST-169-AOT-0300 (C) ; ST-169-AOT-0500 (C) ; ST-169-AOT-0700(C) ; ST-169-AOT-0800 (C) ; ST-169-AOT-0600 (C) ; ST-169-AOTOverSampling-0010 (C) ; ST-169-SacatteringCorr-0010 (C) ; ST-169-Composite-0010 (C) ; ST-169-Composite-0020 (C) ; ST-169-Composite-0030 (C) ; ST-169-Composite-0040 (C) ; ST-169-Composite-0050 (C) ;

7.4.6.12 Journal d'essai de la recette

7.4.6.12.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

7.4.6.12.2 Conclusion du déroulement du test

RAS

7.4.6.12.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

7.4.7 pyTv-S2-L2NOMINAL-011-F-N-SENTINEL2_MUSCATE-MIXRES_CHAIN

7.4.7.1 Titre

Validation fonctionnelle et numérique de la chaîne L2 (Pre-PostProcessing, entrée/sorties) en mode NOMINAL pour des séries temporelles générées avec des DEM à des résolutions différentes. Les produits utilisés sont des SENTINEL2 Muscate.

7.4.7.2 Objectif

L'objectif de cet essai est de valider le fonctionnement global de la chaîne L2 en mode NOMINAL sur des produits SENTINEL2 Muscate générées avec des DEM à des résolutions différentes.

7.4.7.3 Description

Les options de traitement sont :

- Méthode SPECTROTEMPORAL pour l'estimation des aérosols,
- Correction de l'environnement et des pentes,
- Ecriture à la résolution L2,
- activation de la correction des cirrus

La validation se fait sur un contexte avec un DEM à 120m et un produit L2 à 240m.

7.4.7.4 Liste des données d'entrées

Les images utilisées en entrée sont les suivantes :

- SENTINEL2B_20171010-105012-463_L1C_T31TCH_C_V1-0/SENTINEL2B_20171010105012-463_L1C_T31TCH_C_V1-0_MTD_ALL.xml
- SENTINEL2B_20171008-105012-463_L2A_T31TCH_C_V1-0/SENTINEL2B_20171008105012-463_L2A_T31TCH_C_V1-0_MTD_ALL.xml
- S2__TEST_AUX_REFDE2_31TCH_1001.DBL.DIR
- S2__TEST_AUX_REFDE2_31TCH_1001.HDR

Les fichiers GIPPs utilisés en entrée sont les suivants :

- S2A_TEST_GIP_CKEXTL_S_31TJF____10001_20150703_21000101.EEF
- S2A_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.DBL
- S2A_TEST_GIP_CKQLTL_S_31TJF____10005_20150703_21000101.EEF
- S2A_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.DBL.DIR
- S2B_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.DBL.DIR
- S2A_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.DBL
- S2A_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.HDR
- S2A_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.HDR
- S2A_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.DBL
- S2A_TEST_GIP_L2COMM_L_ALLSITES_10008_20150703_21000101.EEF
- S2B_TEST_GIP_CKEXTL_S_31TJF____10001_20150703_21000101.EEF
- S2B_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.DBL.DIR
- S2A_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.DBL

- S2B_TEST_GIP_CKQLTL_S_31TJF____10005_20150703_21000101.EEF
- S2B_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.HDR
- S2A_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.DBL
- S2B_TEST_GIP_L2SMAC_L_ALLSITES_10001_20150703_21000101.EEF
- S2__TEST_GIP_L2SITE_S_31TJF____10001_00000000_99999999.EEF
- S2A_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.DBL.DIR
- S2B_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.DBL
- S2A_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.HDR
- S2B_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.DBL.DIR
- S2A_TEST_GIP_L2SMAC_L_ALLSITES_10001_20150703_21000101.EEF
- S2B_TEST_GIP_L2COMM_L_ALLSITES_10008_20150703_21000101.EEF
- S2B_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.HDR

7.4.7.5 Liste des produits de sortie

Produit SENTINEL2 L2A MUSCATE

7.4.7.6 Prérequis

Il n'y a pas de prérequis.

7.4.7.7 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

7.4.7.8 Epsilon utilisé sur la non regression

0.001

7.4.7.9 Vérifications à effectuer

Le test génère en sortie une image contenant les angles.

7.4.7.10 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R pyTv-S2-L2NOMINAL-011-F-N-SENTINEL2_MUSCATE-MIXRES_CHAIN

7.4.7.11 Exigences

Ce test couvre les exigences suivantes : MACCS-Exigence 640 (C) ; MACCS-Exigence 650 (C) ;

7.4.7.12 Journal d'essai de la recette

7.4.7.12.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

7.4.7.12.2 Conclusion du déroulement du test

RAS

7.4.7.12.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

7.4.8 pyTv-S2-L2NOMINAL-011-F-N-SENTINEL2_MUSCATE-LUTDIRCOR_CHAIN

7.4.8.1 Titre

Validation fonctionnelle et numérique de la chaîne L2 (Pre-PostProcessing, entrée/sorties) avec un produit SENTINEL2 Muscate en mode NOMINAL.

7.4.8.2 Objectif

L'objectif de cet essai est de valider le fonctionnement global de la chaîne L2 en mode NOMINAL sur des produits SENTINEL2 Muscate comportant différentes résolutions. Dans certains cas la méthode de ROY pour la correction directionnelle ne donne pas de bon résultats. Une méthode utilisant des LUT a alors été ajoutée pour traiter ces cas. Ce test permet donc plus particulièrement de vérifier que la chaîne est capable de bien intégrer cette méthode de correction directionnelle basée sur des LUTs.

7.4.8.3 Description

Les options de traitement sont :

- Méthode SPECTROTEMPORAL pour l'estimation des aérosols,
- Correction de l'environnement et des pentes,
- Ecriture à la résolution L2,
- activation de la correction des cirrus
- méthode utilisant des LUTs pour calculer les coefficients de correction directionnelle.

7.4.8.4 Liste des données d'entrées

Les images utilisées en entrée sont les suivantes :

- SENTINEL2B_20171010-105012-463_L1C_T31TCH_C_V1-0/SENTINEL2B_20171010105012-463_L1C_T31TCH_C_V1-0_MTD_ALL.xml
- SENTINEL2B_20171008-105012-463_L2A_T31TCH_C_V1-0/SENTINEL2B_20171008105012-463_L2A_T31TCH_C_V1-0_MTD_ALL.xml
- S2__TEST_AUX_REFDE2_T31TCH_0001.DBL

Les fichiers GIPPs utilisés en entrée sont les suivants :

- S2A_TEST_GIP_CKEXTL_S_31TJF____10001_20150703_21000101.EEF
- S2A_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.DBL
- S2A_TEST_GIP_CKQLTL_S_31TJF____10005_20150703_21000101.EEF
- S2A_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.DBL.DIR
- S2B_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.DBL.DIR
- S2A_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.DBL
- S2A_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.HDR
- S2A_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.HDR
- S2A_TEST_GIP_L2WATV_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.DBL
- S2A_TEST_GIP_L2COMM_L_ALLSITES_10008_20150703_21000101.EEF
- S2B_TEST_GIP_CKEXTL_S_31TJF____10001_20150703_21000101.EEF
- S2B_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.DBL.DIR
- S2A_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_CKQLTL_S_31TJF____10005_20150703_21000101.EEF

- S2B_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.HDR
- S2A_TEST_GIP_L2DIFT_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.DBL
- S2B_TEST_GIP_L2SMAC_L_ALLSITES_10001_20150703_21000101.EEF
- S2__TEST_GIP_L2SITE_S_31TJF____10001_00000000_99999999.EEF
- S2A_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.DBL
- S2B_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.DBL.DIR
- S2B_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.DBL
- S2A_TEST_GIP_L2DIRT_L_CONTINEN_10002_20150703_21000101.HDR
- S2B_TEST_GIP_L2ALBD_L_CONTINEN_10003_20150703_21000101.HDR
- S2B_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.DBL.DIR
- S2A_TEST_GIP_L2SMAC_L_ALLSITES_10001_20150703_21000101.EEF
- S2B_TEST_GIP_L2COMM_L_ALLSITES_10008_20150703_21000101.EEF
- S2B_TEST_GIP_L2TOCR_L_CONTINEN_10002_20150703_21000101.HDR
- S2__TEST_GIP_DIRCOR_L_ALLSITES_50003_20201010_21000101.DBL.DIR
- S2__TEST_GIP_DIRCOR_L_ALLSITES_50003_20201010_21000101.HDR

7.4.8.5 Liste des produits de sortie

Produit SENTINEL2 L2A MUSCATE

7.4.8.6 Prérequis

Il n'y a pas de prérequis.

7.4.8.7 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

7.4.8.8 Epsilon utilisé sur la non regression

0.0001

7.4.8.9 Vérifications à effectuer

Le test génère en sortie une image contenant les angles.

7.4.8.10 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R pyTv-S2-L2NOMINAL-011-F-N-SENTINEL2_MUSCATE-LUTDIRCOR_CHAIN

7.4.8.11 Exigences

Ce test couvre les exigences suivantes : MACCS-Exigence 640 (C) ; MACCS-Exigence 650 (C) ;

7.4.8.12 Journal d'essai de la recette

7.4.8.12.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

7.4.8.12.2 Conclusion du déroulement du test

RAS

7.4.8.12.3 Validation du test

Date de validation	Résultat
03/11/2020	OK

7.4.9 pyTv-S2-L2INIT-CROSSWRITING-SENTINEL2-SENTINEL2MUSCATE

7.4.9.1 Titre

Validation fonctionnelle et performances de la chaîne L2 en mode INIT avec un produit SENTINEL2 Natif mais produisant un SENTINEL2 Muscate (mode INIT).

7.4.9.2 Objectif

L'objectif de ce test est de valider le traitement en mode INIT d'un produit SENTINEL2 Natif et de produire en sortie un produit SENTINEL2 Muscate.

7.4.9.3 Description

Instance INIT de SENTINEL2 Natif afin de produire un SENTINEL2 Muscate

7.4.9.4 Liste des données d'entrées

Produits:

- S2__TEST_AUX_REFDE2_36JTT____5001.DBL
- S2__TEST_AUX_REFDE2_36JTT____5001.HDR
- S2A_MSIL1C_20160914T074612_N0204_R135_T36JTT_20160914T081456.SAFE

Gipps

- S2A_TEST_GIP_L2DIFT_L_CONTINEN_50003_20150703_21000101.DBL
- S2A_TEST_GIP_L2TOCR_L_CONTINEN_50003_20150703_21000101.DBL.DIR
- S2__TEST_EXO_METDTA_20090101T000000_20111231T230000.DBL
- S2A_TEST_GIP_CKEXTL_S_36JTT____50001_00000000_99999999.EEF
- S2A_TEST_GIP_L2DIFT_L_CONTINEN_50003_20150703_21000101.DBL.DIR
- S2A_TEST_GIP_L2TOCR_L_CONTINEN_50003_20150703_21000101.HDR
- S2__TEST_EXO_METDTA_20090101T000000_20111231T230000.DBL.BIN
- S2A_TEST_GIP_CKQLTL_S_36JTT____50001_00000000_99999999.EEF
- S2A_TEST_GIP_L2DIFT_L_CONTINEN_50003_20150703_21000101.HDR
- S2A_TEST_GIP_L2WATV_L_CONTINEN_50003_20150703_21000101.DBL
- S2__TEST_EXO_METDTA_20090101T000000_20111231T230000.HDR
- S2A_TEST_GIP_L2ALBD_L_CONTINEN_50004_20150703_21000101.DBL
- S2A_TEST_GIP_L2DIRT_L_CONTINEN_50003_20150703_21000101.DBL
- S2A_TEST_GIP_L2WATV_L_CONTINEN_50003_20150703_21000101.DBL.DIR
- S2__TEST_GIP_L2SITE_S_36JTT____50001_00000000_99999999.EEF
- S2A_TEST_GIP_L2ALBD_L_CONTINEN_50004_20150703_21000101.DBL.DIR
- S2A_TEST_GIP_L2DIRT_L_CONTINEN_50003_20150703_21000101.DBL.DIR
- S2A_TEST_GIP_L2WATV_L_CONTINEN_50003_20150703_21000101.HDR
- S2A_TEST_GIP_L2ALBD_L_CONTINEN_50004_20150703_21000101.HDR
- S2A_TEST_GIP_L2DIRT_L_CONTINEN_50003_20150703_21000101.HDR
- S2A_TEST_GIP_L2SMAC_L_ALLSITES_50003_20150703_21000101.EEF
- S2__TEST_AUX_REFDE2_36JTT____5001.DBL.DIR
- S2A_TEST_GIP_L2COMM_L_ALLSITES_50008_20150703_21000101.EEF
- S2A_TEST_GIP_L2TOCR_L_CONTINEN_50003_20150703_21000101.DBL

7.4.9.5 Liste des produits de sortie

Produit L2A SENTINEL2 Muscate

7.4.9.6 Prérequis

Il n'y a pas de prérequis.

7.4.9.7 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

7.4.9.8 Epsilon utilisé sur la non regression

0.0001

7.4.9.9 Vérifications à effectuer

Les tests COMP_ASCII et COMP_IMAGE associés permettent de valider la non regression

7.4.9.10 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : `ctest -R pyTv-S2-L2INIT-CROSSWRITING-SENTINEL2-SENTINEL2MUSCATE`

7.4.9.11 Exigences

Ce test couvre les exigences suivantes : Néant

7.4.9.12 Journal d'essai de la recette

7.4.9.12.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

7.4.9.12.2 Conclusion du déroulement du test

RAS

7.4.9.12.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

7.4.10 pyTv-S2-L2INIT-CROSSWRITING-LANDSAT8-LANDSAT8MUSCATE

7.4.10.1 Titre

Validation fonctionnelle et performances de la chaîne L2 en mode INIT avec un produit LANDSAT8 Natif mais produisant un LANDSAT8 Muscate (mode INIT).

7.4.10.2 Objectif

L'objectif de ce test est de valider le traitement en mode INIT d'un produit LANDSAT8 Natif et de produire en sortie un produit LANDSAT8 Muscate.

7.4.10.3 Description

Instance INIT de LANDSAT8 Natif afin de produire un LANDSAT8 Muscate

7.4.10.4 Liste des données d'entrées

Produits:

- L8_TEST_AUX_REFDE2_199029_0001.DBL
- L8_TEST_AUX_REFDE2_199029_0001.DBL.DIR
- L8_TEST_AUX_REFDE2_199029_0001.HDR
- LC08_L1TP_199029_20170527_20170615_01_T1

Gipps:

- L8_TEST_GIP_CKQLTL_S_EU93056200A00B_50001_00000000_99999999.EEF
- L8_TEST_GIP_L2DIFT_L_CONTINEN_50001_00000000_99999999.HDR
- L8_TEST_GIP_L2TOCR_L_CONTINEN_50001_00000000_99999999.DBL.DIR
- L8_TEST_GIP_L2ALBD_L_CONTINEN_50001_00000000_99999999.DBL
- L8_TEST_GIP_L2DIRT_L_CONTINEN_50001_00000000_99999999.DBL
- L8_TEST_GIP_L2TOCR_L_CONTINEN_50001_00000000_99999999.HDR
- L8_TEST_GIP_L2ALBD_L_CONTINEN_50001_00000000_99999999.DBL.DIR
- L8_TEST_GIP_L2DIRT_L_CONTINEN_50001_00000000_99999999.DBL.DIR
- L8_TEST_EXO_METDTA_20090101T000001_20111231T230000.DBL
- L8_TEST_GIP_L2ALBD_L_CONTINEN_50001_00000000_99999999.HDR
- L8_TEST_GIP_L2DIRT_L_CONTINEN_50001_00000000_99999999.HDR
- L8_TEST_EXO_METDTA_20090101T000001_20111231T230000.DBL.BIN

- L8_TEST_GIP_L2COMM_L_ALLSITES_50001_00000000_99999999.EEF
- L8_TEST_GIP_L2SITE_S_EU93056200A00B_50001_00000000_99999999.EEF
- L8_TEST_EXO_METDTA_20090101T000001_20111231T230000.HDR
- L8_TEST_GIP_L2DIFT_L_CONTINEN_50001_00000000_99999999.DBL
- L8_TEST_GIP_L2SMAC_L_ALLSITES_50001_00000000_99999999.EEF
- L8_TEST_GIP_CKEXTL_S_EU93056200A00B_50001_00000000_99999999.EEF
- L8_TEST_GIP_L2DIFT_L_CONTINEN_50001_00000000_99999999.DBL.DIR
- L8_TEST_GIP_L2TOCR_L_CONTINEN_50001_00000000_99999999.DBL

7.4.10.5 Liste des produits de sortie

Produit L2A LANDSAT8 Muscate

7.4.10.6 Prérequis

Il n'y a pas de prérequis.

7.4.10.7 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

7.4.10.8 Epsilon utilisé sur la non regression

0.0001

7.4.10.9 Vérifications à effectuer

Les tests COMP_ASCII et COMP_IMAGE associés permettent de valider la non regression

7.4.10.10 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : `ctest -R pyTv-S2-L2INIT-CROSSWRITING-LANDSAT8-LANDSAT8MUSCATE`

7.4.10.11 Exigences

Ce test couvre les exigences suivantes : Néant

7.4.10.12 Journal d'essai de la recette

7.4.10.12.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

7.4.10.12.2 Conclusion du déroulement du test

RAS

7.4.10.12.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

7.4.11 pyTv-S2-L2INIT-CROSSWRITING-VENUS-VENUSMUSCATE

7.4.11.1 Titre

Validation fonctionnelle et performances de la chaîne L2 en mode INIT avec un produit VENUS Natif mais produisant un VENUS Muscate (mode INIT).

7.4.11.2 Objectif

L'objectif de ce test est de valider le traitement en mode INIT d'un produit VENUS Natif et de produire en sortie un produit VENUS Muscate.

7.4.11.3 Description

Instance INIT de VENUS Natif afin de produire un VENUS Muscate

7.4.11.4 Liste des données d'entrées

Produits:

- VE_TEST_AUX_REFDE2_CAMARGUE_0001.DBL.DIR
- VE_TEST_AUX_REFDE2_CAMARGUE_0001.HDR
- VE_VM01_VSC_L1VALD__PE_R_U__20120101.DBL.DIR
- VE_VM01_VSC_L1VALD__PE_R_U__20120101.HDR

Gipps

- VE_TEST_GIP_CKQLTL_S_CAMARGUE_00001_00000000_99999999.EEF
- VE_TEST_GIP_L2DIRT_L_CONTINEN_00001_00000000_99999999.DBL.DIR
- VE_TEST_GIP_L2WATV_L_CONTINEN_00001_00000000_99999999.DBL.DIR
- VE_TEST_GIP_L2ALBD_L_CONTINEN_00001_00000000_99999999.DBL.DIR

- VE_TEST_GIP_L2DIRT_L_CONTINEN_00001_00000000_99999999.HDR
- VE_TEST_GIP_L2WATV_L_CONTINEN_00001_00000000_99999999.HDR
- VE_TEST_GIP_L2ALBD_L_CONTINEN_00001_00000000_99999999.HDR
- VE_TEST_GIP_L2SITE_S_CAMARGUE_00001_00000000_99999999.EEF
- VE_TEST_EXO_METDTA_20110101T000000_20111231T230000.DBL.BIN
- VE_TEST_GIP_L2COMM_L_ALLSITES_00001_00000000_99999999.EEF
- VE_TEST_GIP_L2SMAC_L_ALLSITES_00001_00000000_99999999.EEF
- VE_TEST_EXO_METDTA_20110101T000000_20111231T230000.HDR
- VE_TEST_GIP_L2DIFT_L_CONTINEN_00001_00000000_99999999.DBL.DIR
- VE_TEST_GIP_L2TOCR_L_CONTINEN_00001_00000000_99999999.DBL.DIR
- VE_TEST_GIP_CKEXTL_S_CAMARGUE_00001_00000000_99999999.EEF
- VE_TEST_GIP_L2DIFT_L_CONTINEN_00001_00000000_99999999.HDR
- VE_TEST_GIP_L2TOCR_L_CONTINEN_00001_00000000_99999999.HDR

7.4.11.5 Liste des produits de sortie

Produit L2A Venus Muscate

7.4.11.6 Prérequis

Il n'y a pas de prérequis.

7.4.11.7 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

7.4.11.8 Epsilon utilisé sur la non regression

0.0001

7.4.11.9 Vérifications à effectuer

Les tests COMP_ASCII et COMP_IMAGE associés permettent de valider la non regression

7.4.11.10 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : `ctest -R pyTv-S2-L2INIT-CROSSWRITING-VENUS-
VENUSMUSCATE`

7.4.11.11 Exigences

Ce test couvre les exigences suivantes : Néant

7.4.11.12 Journal d'essai de la recette

7.4.11.12.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

7.4.11.12.2 Conclusion du déroulement du test

RAS

7.4.11.12.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

7.4.12 pyTv-VE-L2INIT-VENUS-MUSCATE_CHAIN

7.4.12.1 Titre

Validation fonctionnelle et numérique de la chaîne L2 (Pre-PostProcessing, entrée/sorties) avec un produit VENUS Muscate en mode INIT.

7.4.12.2 Objectif

L'objectif de cet essai est de valider le fonctionnement global de la chaîne L2 en mode INIT sur des produits VENUS Muscate. Ce test permet également de tester le bon déroulement de la chaîne lorsque l'écriture des données SRE est désactivée.

7.4.12.3 Description

Les options de traitement sont :

- Instance INIT de VENUS Muscate afin de produire un VENUS Muscate
- Désactivation de l'écriture des images SRE
- Activation de la correction environnementale

7.4.12.4 Liste des données d'entrées

Les images utilisées en entrée sont les suivantes :

- VENUS-XS_20120103-105148-000_L1C_CAMARGUE_C_V1-0/VENUS-XS_20120103-105148-000_L1C_CAMARGUE_C_V1-0_MTD_ALL.xml
- VE_TEST_AUX_REFDE2_CAMARGUE_0002.DBL
- VE_TEST_AUX_REFDE2_CAMARGUE_0002.HDR
- VE_TEST_AUX_REFDE2_CAMARGUE_0002.DBL.DIR

Les fichiers GIPPs utilisés en entrée sont les suivants :

-VE_TEST_EXO_METDTA_20110101T000001_20111231T230000.DBL -VE_TEST_EXO_METDTA_20110101T000001_20111231T230000.DBL
-VE_TEST_GIP_CKEXTL_S_CAMARGUE_00005_00000000_99999999.EEF -VE_TEST_GIP_CKQTL_S_CAMARGUE_00002_00000000_99999999.DBL
-VE_TEST_GIP_L2ALBD_L_CONTINEN_00002_00000000_99999999.DBL -VE_TEST_GIP_L2ALBD_L_CONTINEN_00002_00000000_99999999.HDR
-VE_TEST_GIP_L2ALBD_L_CONTINEN_00002_00000000_99999999.HDR -VE_TEST_GIP_L2COMM_L_ALLSITES_00001_20110101T000001_20111231T230000.DBL
-VE_TEST_GIP_L2DIFT_L_CONTINEN_00002_00000000_99999999.DBL -VE_TEST_GIP_L2DIFT_L_CONTINEN_00002_00000000_99999999.HDR
-VE_TEST_GIP_L2DIFT_L_CONTINEN_00002_00000000_99999999.HDR -VE_TEST_GIP_L2DIRT_L_CONTINEN_00002_00000000_99999999.DBL
-VE_TEST_GIP_L2DIRT_L_CONTINEN_00002_00000000_99999999.DBL.DIR -
VE_TEST_GIP_L2DIRT_L_CONTINEN_00002_00000000_99999999.HDR -VE_TEST_GIP_L2SITE_S_CAMARGUE_00002_00000000_99999999.DBL
-VE_TEST_GIP_L2SMAC_L_ALLSITES_00002_00000000_99999999.EEF -VE_TEST_GIP_L2TOCR_L_CONTINEN_00002_00000000_99999999.DBL
-VE_TEST_GIP_L2TOCR_L_CONTINEN_00002_00000000_99999999.DBL.DIR -
VE_TEST_GIP_L2TOCR_L_CONTINEN_00002_00000000_99999999.HDR -VE_TEST_GIP_L2WATV_L_CONTINEN_00002_00000000_99999999.DBL
-VE_TEST_GIP_L2WATV_L_CONTINEN_00002_00000000_99999999.DBL.DIR -
VE_TEST_GIP_L2WATV_L_CONTINEN_00002_00000000_99999999.HDR

7.4.12.5 Liste des produits de sortie

Produit VENUS MUSCATE L2A

7.4.12.6 Prérequis

Il n'y a pas de prérequis.

7.4.12.7 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

7.4.12.8 Epsilon utilisé sur la non regression

0.001

7.4.12.9 Vérifications à effectuer

Le test génère en sortie une image contenant les angles.

7.4.12.10 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : `ctest -R pyTv-VE-L2INIT-VENUS-MUSCATE_CHAIN`

7.4.12.11 Exigences

Ce test couvre les exigences suivantes : MACCS-Exigence 640 (C) ; MACCS-Exigence 650 (C) ;

7.4.12.12 Journal d'essai de la recette

7.4.12.12.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

7.4.12.12.2 Conclusion du déroulement du test

RAS

7.4.12.12.3 Validation du test

Date de validation	Résultat
25/11/2020	OK

7.4.13 pyTv-STARTMAJA_CHAIN

7.4.13.1 Titre

Validation fonctionnelle et numérique de StartMaja avec des produits SENTINEL2 en modes INIT et NOMINAL.

7.4.13.2 Objectif

L'objectif de cet essai est de valider le fonctionnement global de StartMaja en mode INIT ET NOMINAL sur des produits SENTINEL2. Plus particulièrement, ce test permet de vérifier que la chaîne est bien capable de télécharger les GIPPS et autres données nécessaires, et de bien faire le lien avec l'orchestrateur MAJA.

7.4.13.3 Description

Les options de traitement sont :

- Méthode SPECTROTEMPORAL pour l'estimation des aérosols,
- Correction de l'environnement et des pentes,
- Ecriture à la résolution L2,
- activation de la correction des cirrus

7.4.13.4 Liste des données d'entrées

Les images utilisées en entrée sont les suivantes :

- S2A_MSIL1C_20170605T105031_N0205_R051_T31UDQ_20170605T105303.SAFE
- S2A_MSIL1C_20170615T105031_N0205_R051_T31UDQ_20170615T105505.SAFE
- S2__TEST_AUX_REFDE2_31UDQ_0001.DBL.DIR
- S2__TEST_AUX_REFDE2_31UDQ_0001.HDR

Les fichiers GIPPs utilisés en entrée sont les suivants :

- S2__TEST_EXO_CAMS_20170625T000000_20190418T025513.DBL.DIR
- S2__TEST_EXO_CAMS_20170625T000000_20190418T025513.HDR
- S2__TEST_EXO_CAMS_20170625T120000_20190418T053109.DBL.DIR
- S2__TEST_EXO_CAMS_20170625T120000_20190418T053109.HDR

7.4.13.5 Liste des produits de sortie

Produit SENTINEL2 L2A MUSCATE

7.4.13.6 Prérequis

Il n'y a pas de prérequis.

7.4.13.7 Durée attendue

La durée d'exécution de l'essai n'est pas un critère attendu.

7.4.13.8 Epsilon utilisé sur la non regression

0.0001

7.4.13.9 Vérifications à effectuer

Le test génère en sortie une image contenant les angles.

7.4.13.10 Mise en oeuvre du test

Ce test est exécuté en lançant la commande : ctest -R pyTv-STARTMAJA_MAJA

7.4.13.11 Exigences

Ce test couvre les exigences suivantes : MACCS-Exigence 640 (C) ; MACCS-Exigence 650 (C) ;

7.4.13.12 Journal d'essai de la recette

7.4.13.12.1 Notes sur le déroulement du test

Rien de particulier n'a été noté lors du déroulement du test.

7.4.13.12.2 Conclusion du déroulement du test

RAS

7.4.13.12.3 Validation du test

Date de validation	Résultat
26/11/2010	OK

ANNEXES

BIBLIOGRAPHY

- [LD] Liste documentaire, SETG-LD-MAJA-010-CS
- [SDM] Software Developer Manual, SETG-MD-MAJA-010-CS
- [MU] Manuel utilisateur (anglais) de Maja, SETG-MU-MAJA-010-CS
- [PSD] Interfaces des produits Sentinel2, S2-PDGS-TAS-DI-PSD
- [PSC] Interfaces des produits Muscate, PSC-SL-411-0032
- [DA01] Spécification technique du besoin, SETG-STB-MAJA-0636-CNES
- [DA02] Spécification of MACCS-ATCOR joint algorithms for level 2A product and of Venus level 3 product, MAJA-TN-WP1-008-CNES
- [DA06] Technical specifications, Venus products specifications, VE-ST-GSSM-39-CNES
- [DA07] Technical notes, Venus ground segment interfaces, VE-NT-GSSM-196-CNES
- [DA100] Document de conception globale des chaines L2 et L3 Venus, VE-CG-GSSM-010-CS