RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

CNES
CENTRE NATIONAL D'ÉTUDES SPATIALES

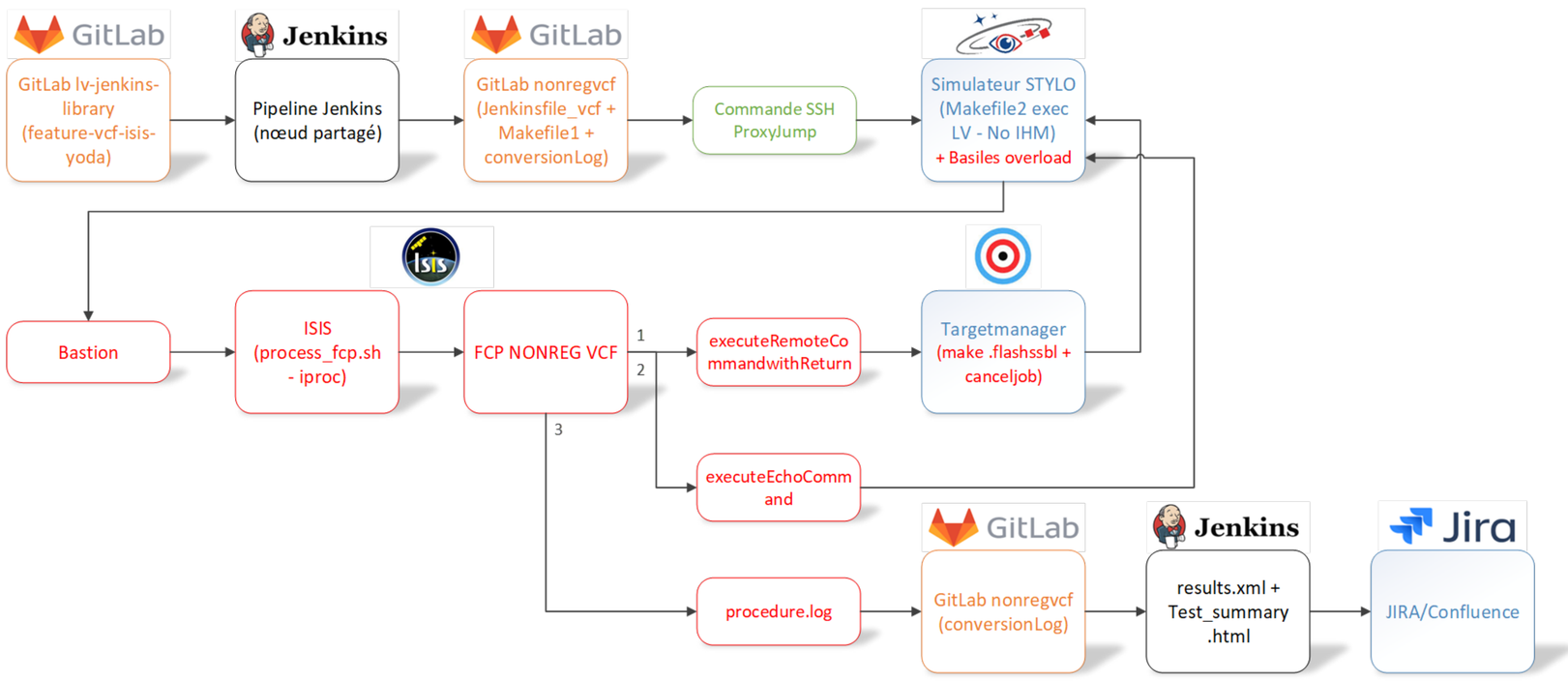
VCF automatique
pipeline de non-regression pour valider LV, BDS et procédures

Guillaume VANNESSON (TSN) et Camille SOUCHET (CNES)

OBJECTIFS

La pipeline d'automatisation des tests VCF a pour objectifs de :
- Construire le logiciel de vol, préparer l'environnement de test et lancer les campagnes de non-régression VCF
- Valider rapidement les modifications **LV, BDS et FCP** pour la **VCF, l'AIT et OP**
- Répondre aux besoins budgets/plannings des projets de plus en plus restreints

Acronymes
LV: Logiciel de Vol
BDS : Base de Données Satellite
FCP : Flight Control Procedure
ISIS : le centre de contrôle
STYLO : le simulateur



RESULTATS

Le procedure.log – fichier de log d'exécution des procédures générées par le centre de contrôle – est extrait et interprété pour récupérer l'ensemble des informations permettant de savoir quelles procédures sont jouées, avec quels arguments et dans le cadre de quel test. Ces informations sont ensuite synthétisées dans un fichier .xml qui est affichable sous le format de page web.

Test suite	Test case	Arguments	last build (19)	18	17	16
CL-VCF-NONREG-AUTO						
NONREG	CL-VCF-NONREG		success	success	success	fail
CLMGT-VCF-0101-01	CL-POWER-ON		success	success	success	success
CLMGT-VCF-0101-01	CL-POWER-OFF		success	success	success	success
CLMGT-VCF-0101-01	CL-TESTCONNECT		success	success	success	success
CLMGT-VCF-0101-03	CL-SETCONF-WA	{'arg1': 'clconf', 'val1': 'DET_AVANCEE'}	success	success	success	success
CLMGT-VCF-0101-04	CL-GETREGISTER-ALL		success	success	success	success
CLMGT-VCF-0101-05	CL-GETPARAMVAL-ALL		success	success	success	success
CLMGT-VCF-0101-05	CL-SETPARAMVAL-CL_AJ_PICSIZE-1024-CENTER		success	success	success	success
CLMGT-VCF-0101-05	CL-GETPARAMVAL-CL_AJ_PICSIZE-1024-CENTER		success	success	success	success
CLMGT-VCF-0101-04	CL-SETREGISTER-CL_AJ_PICSIZE-DEFAULT-WA		success	success	success	success
CLMGT-VCF-0101-04	CL-SETREGISTER-NBDATA2-WA	{'arg1': 'regdata1', 'val1': '0x00', 'arg2': 'regdata2', 'val2': '0x00', 'arg3': 'startaddr', 'val3': '0x15'}	success	success	success	success
CLMGT-VCF-0101-04	CL-GETREGISTER-CL_AJ_PICSIZE-DEFAULT-WA		success	success	success	success
CLMGT-VCF-0101-04	CL-GETREGISTER-WA	{'arg1': 'nbdata', 'val1': '2', 'arg2': 'startaddr', 'val2': '0x15'}	success	success	success	success
CLMGT-VCF-0101-02	CL-PICTURETOGROUND-WA	{'arg1': 'inttime', 'val1': '10', 'arg2': 'picid', 'val2': '1'}	success	success	success	success
CLMGT-VCF-0101-02	CL-PICTURETOGYSELE-WA	{'arg1': 'inttime', 'val1': '10', 'arg2': 'picid', 'val2': '2', 'arg3': 'picsave', 'val3': 'OUT'}	success	success	success	fail
CLMGT-VCF-0101-02	CL-PICTURETOFUGY-WA	{'arg1': 'inttime', 'val1': '10', 'arg2': 'picid', 'val2': '4', 'arg3': 'picsave', 'val3': 'OUT'}	success	success	success	none

Figure 2 : Sortie au format HTML du détails des procédures jouées pendant la séquence de tests

La base de données JIRA permet de définir les tests, de suivre les dépendances et la bonne complétion des actions nécessaires au lancement des tests (développements de procédures, autres tests à réaliser) ainsi que de visualiser la réalisation des tests et leurs résultats.
Ci-dessous un exemple de ticket de test où on retrouve la liste des procédures jouées et les développements nécessaires au déroulement du tests avec leurs statuts d'avancement.

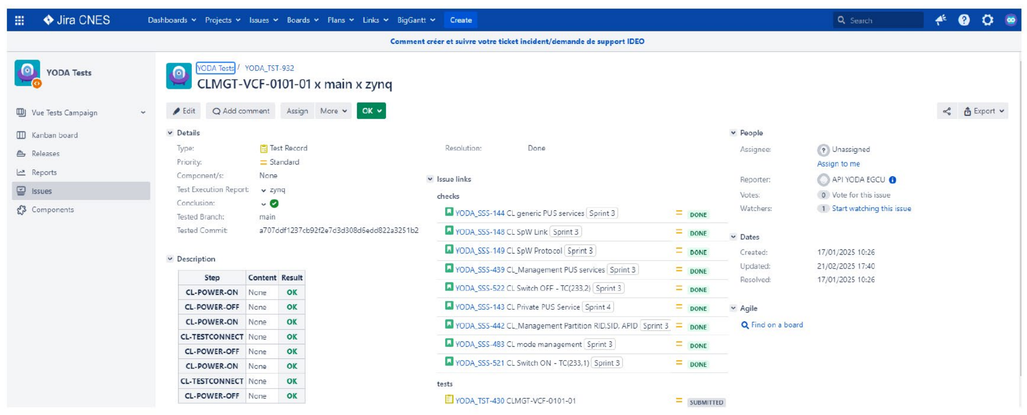


Figure 3 : Base de données JIRA répertoriant les séquences de test et les procédures

JIRA permet d'afficher clairement les résultats des compagnes de validation. Plusieurs dashboards spécifiques ont été créés pour visualiser:
- dans le cas de la Figure 4, le statut de chaque test, sa dernière mise à jour et les procédures jouées (description)
- dans le cas de la Figure 5, le résumé, le statut, les différentes dates ainsi que la résolution du ticket.

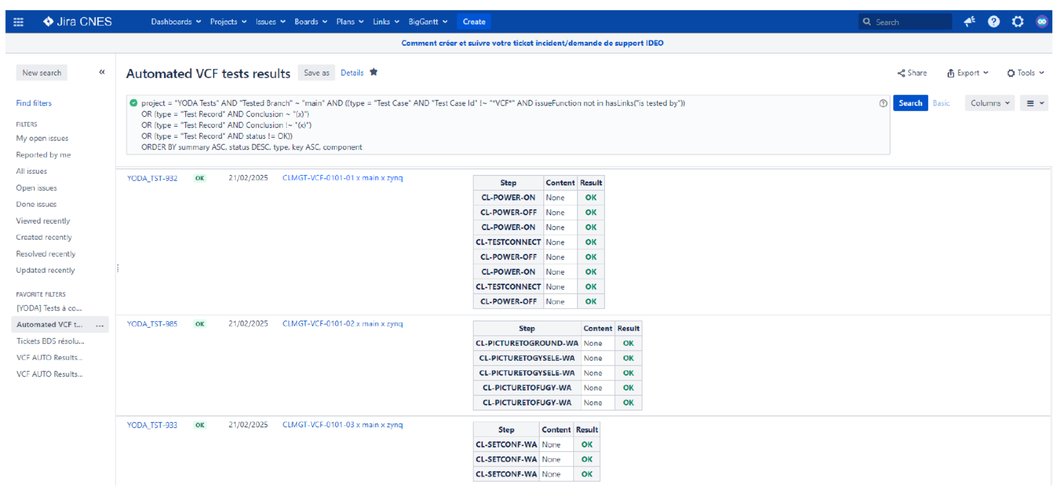


Figure 4 : Dashboard des résultats des tests VCF avec description

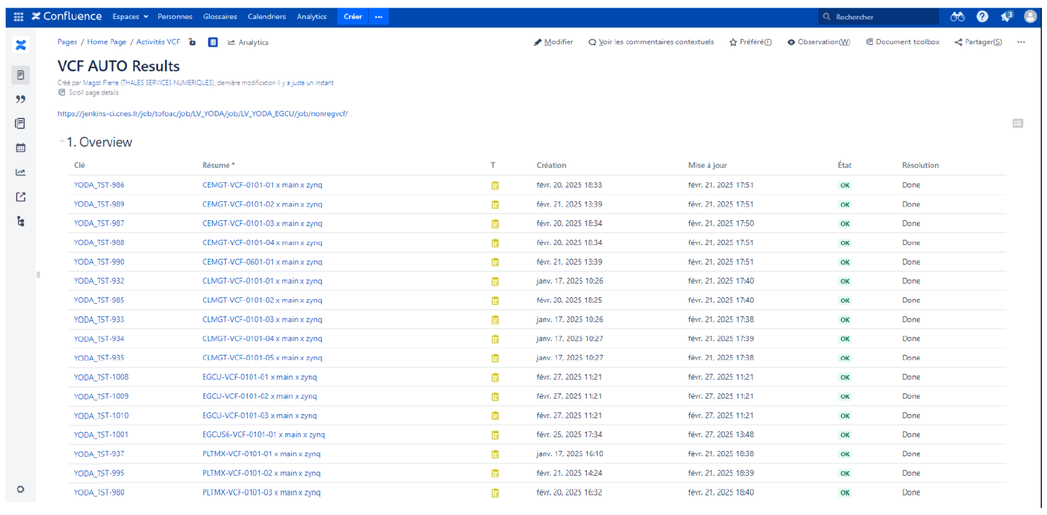


Figure 5 : Dashboard des résultats des tests VCF

CONCLUSIONS

- La VCF YODA via Jenkins avec la ligne de produit ISIS intégrée au cœur de l'automatisation est fonctionnelle
- Deux campagnes de non-regression du LV et de la BDS ont été réalisées
- Elle apporte un gain de temps important car les tests sont jouées hors heures ouvrées

POUR ALLER PLUS LOIN

Migration de Jenkins vers Gitlab-CI à réaliser dès que possible
Rendre la pipeline générique sur la ligne de produit ISIS pour utiliser ce type de pipeline aux différentes étapes du projet – développement (VCF, AIT) et en exploitation (opérations)

THALES

Présenté aux
RENCONTRES TECHNIQUES ET NUMERIQUES
2 décembre 2025
Toulouse, Centre de Congrès Pierre Baudis