

COMPTE RENDU DE STAGE

STAGE DE 1ÈRE ANNÉE — BTS SIO SLAM

Cookson-CLAL

Dardilly (69) · Groupe HM Precious Metals

Maxime Blanco

Lycée Lamartine — Mâcon · BTS SIO option SLAM

Mai — Juillet 2025

Sommaire

- 01 — Présentation de l'entreprise**
- 02 — Contexte et objectif du stage
- 03 — Environnement technique
- 04 — Déroulé des travaux
 - 01 · Compréhension et traduction du script
 - 02 · Phase de test
 - 03 · Mise en place et déploiement
- 05 — Compétences BTS SIO validées
- 06 — Bilan personnel

01 — Présentation de l'entreprise

Cookson-CLAL en bref

Cookson-CLAL est une entreprise spécialisée dans l'achat, la vente et la transformation de métaux précieux (or, argent, platine) ainsi que dans la fourniture d'outillage professionnel pour les bijoutiers. Elle s'adresse à la fois à des industriels et à des artisans du secteur de la bijouterie-joaillerie.

L'entreprise fait partie du groupe international HM Precious Metals, dont le siège social est situé en Allemagne. Le site de Dardilly (69) concentre les activités commerciales et de service client pour la France.

Fiche d'identité

| | |
|----------------|--|
| Raison sociale | Cookson-CLAL |
| Secteur | Industrie — Métaux précieux & bijouterie |
| Groupe | HM Precious Metals (Siège Allemagne) |
| Adresse | 69570 Dardilly (Rhône) |
| Site web | www.cookson-clal.com |
| Activité IT | Service informatique interne, gestion du parc Windows |

Organisation IT

Le service informatique de Cookson-CLAL est composé d'un administrateur systèmes & réseaux qui gère l'ensemble du parc informatique : postes Windows, serveurs, réseau, et les outils métier spécifiques au secteur (AS400, LANSA). C'est dans ce service que j'ai effectué mon stage.

02 — Contexte et objectif du stage

Problématique identifiée

Lors de la prise de poste, le script de connexion réseau de l'entreprise — exécuté automatiquement à chaque ouverture de session Windows — présentait plusieurs dysfonctionnements dus à l'obsolescence progressive du VBScript.

Problèmes constatés :

- Le script VBScript de login était partiellement obsolète
- Certaines applications et automatisations ne fonctionnaient plus correctement
- Les utilisateurs devaient effectuer des manipulations manuelles supplémentaires au démarrage
- Le service informatique était surchargé par des tickets liés à des problèmes mineurs de session
- Des interventions importantes étaient interrompues par des utilisateurs bloqués faute de session stable

Objectif assigné

Ma mission principale consistait à réécrire le script de connexion réseau en PowerShell, en remplacement du VBScript obsolète. Ce script est critique pour l'entreprise car il s'exécute au démarrage de session pour l'ensemble des employés, configurant :

- Les lecteurs réseau partagés (par département : Finance, Commerce, Informatique, Laboratoire, etc.)
- Les applications à lancer automatiquement selon le profil utilisateur
- Les autorisations Active Directory associées à chaque groupe
- La signature de mail Outlook générée automatiquement

Deux sous-objectifs accompagnaient cette mission :

- Apprendre et comprendre les deux langages concernés (VBScript et PowerShell) ainsi que les besoins spécifiques de l'entreprise
- Trier, moderniser et ajouter des fonctionnalités nouvelles au script pour chaque groupe d'employés

03 — Environnement technique

Outils et langages en place dans l'entreprise

L'environnement informatique de Cookson-CLAL s'appuie sur un ensemble de technologies à la fois historiques et modernes, caractéristique des entreprises industrielles de sa génération.

| Technologie | Origine | Description |
|-------------------------|------------------|--|
| AS400 | IBM — 1988 | ERP historique utilisé pour la gestion des stocks et des commandes de métaux précieux. Interface textuelle via terminal, toujours en production. |
| LANSA | IBM — post AS400 | Langage et environnement de développement lié à IBM, utilisé en complément de l'AS400 pour les interfaces plus récentes. |
| VBScript | Microsoft — 1991 | Langage de script Microsoft utilisé historiquement pour le script de login Windows. Partiellement obsolète, en cours de remplacement. |
| PowerShell | Microsoft — 2006 | Langage de scripting moderne pour l'automatisation des tâches Windows. Remplaçant naturel du VBScript, activement maintenu par Microsoft. |
| Active Directory | Microsoft | Annuaire d'entreprise gérant les comptes utilisateurs, groupes, autorisations et politiques de sécurité (GPO). |
| Windows Server | Microsoft | Système d'exploitation serveur hébergeant les partages réseau, l'Active Directory et les services métier. |

04 — Déroulé des travaux

Vue d'ensemble de la méthode

Le travail a été organisé en quatre phases successives, chacune validée avant de passer à la suivante :

- Organisation — Planification des tâches et prise de connaissance de l'environnement
- Compréhension / Traduction — Analyse et réécriture du script
- Phase de test — Validation sur machines virtuelles et postes physiques
- Mise en place — Déploiement sur les postes utilisateurs en production

01 · Compréhension et traduction du script

La première étape consistait à apprendre et comprendre deux langages : le VBScript (langage du script existant) et le PowerShell (langage cible). J'ai commencé par analyser en profondeur la logique du script VBScript pour en comprendre toutes les fonctionnalités.

Les deux langages présentent de nombreuses similarités structurelles (conditions, boucles, gestion des objets COM), ce qui a facilité la traduction. Le script gère notamment le montage des lecteurs réseau par département via des instructions Switch/Case :

```
VBScript (original)
Select Case tab(j)
Case "s_finances"
    $network.MapNetworkDrive "U:",
    service&"\Finances"
    $ObjShell.NameSpace("U:").Self.Name =
    "Finances"
```

```
PowerShell (traduit)
switch ($x) {
    "s_finances" {
        $network.MapNetworkDrive("U:",
        "$service\Finances")
        $ObjShell.NameSpace("U:").Self.Name =
        "Finances"
```

02 · Phase de test

Une fois le script réécrit, j'ai procédé à une phase de tests rigoureux, indispensable compte tenu de l'impact sur l'ensemble des postes de l'entreprise.

Objectifs des tests :

- Tester chaque groupe d'utilisateurs (Finance, Commerce, Informatique, Laboratoire, Lyon, Management, Mill, Marseille, Afinor, etc.)
- Vérifier la fiabilité du montage des lecteurs réseau pour chaque profil
- Vérifier les autorisations et accès Active Directory
- S'assurer que les applications se lancent correctement

Méthodologie :

- Phase 1 : Tests sur machine virtuelle isolée pour débugger sans risque de production

- Phase 2 : Tests sur un poste physique personnel en conditions réelles
- Phase 3 : Tests croisés sur plusieurs profils utilisateurs

03 · Mise en place et déploiement

La dernière étape consistait à rendre le script fonctionnel au démarrage de session Windows via les Group Policy Objects (GPO) et à le déployer pour les utilisateurs réels.

Actions réalisées :

- Configuration du script pour s'exécuter automatiquement à l'ouverture de session via PowerShell et les GPO Windows
- Test des fonctionnalités sur des postes utilisateurs réels (différents services)
- Vérification et configuration de la signature de mail Outlook — générée automatiquement avec le nom, poste, adresse et logo Cookson-CLAL de l'utilisateur connecté
- Ajustements et corrections suite aux retours des premiers tests en production

05 — Compétences BTS SIO validées

Blocs de compétences acquises

| | |
|-----|--|
| 1.1 | Recenser et identifier les ressources numériques — Inventaire du parc logiciel et des scripts existants |
| 1.2 | Exploiter des référentiels, normes et standards — Compréhension et application des standards PowerShell et GPO |
| 2.2 | Collecter, suivre et orienter des demandes — Analyse des besoins utilisateurs liés aux dysfonctionnements |
| 2.3 | Traiter des demandes concernant les applications — Résolution des problèmes d'applications au démarrage de session |
| 5.1 | Réaliser les tests d'intégration et d'acceptation — Tests par groupes utilisateurs sur VM et postes physiques |
| 6.1 | Mettre en place son environnement d'apprentissage — Apprentissage autonome de PowerShell en contexte professionnel |

06 — Bilan personnel

Ce que ce stage m'a apporté

Ce stage de première année m'a permis de découvrir concrètement comment l'informatique s'intègre dans une entreprise industrielle, avec des contraintes et des enjeux réels très différents du contexte scolaire.

Compétences techniques acquises

- Maîtrise de PowerShell pour l'automatisation et la gestion Windows
- Lecture et compréhension de VBScript (code existant en production)
- Compréhension du fonctionnement de l'Active Directory et des GPO
- Découverte de l'AS400 et de LANSA, outils spécifiques au secteur industriel
- Travail sur un environnement Windows Server en conditions réelles

Compétences transversales

- Rigueur et méthode : chaque modification du script impactant tous les postes, les tests devaient être exhaustifs
- Autonomie : j'ai dû apprendre deux langages de façon indépendante pour mener à bien la mission
- Responsabilité : travailler sur un script critique en production m'a appris la prudence et la rigueur en environnement professionnel
- Collaboration : échanges réguliers avec l'administrateur systèmes pour comprendre les besoins et valider les livrables

Conclusion

Ce premier stage m'a confirmé mon intérêt pour le développement et l'informatique en entreprise. Bien que la mission était principalement orientée systèmes (scripts, administration Windows), elle m'a permis de développer une rigueur de travail et une logique de programmation applicables à tous mes futurs projets, y compris en développement front-end.

La transition vers le stage de 2ème année chez Xefi m'a permis de mettre à profit ces bases solides pour aborder des technologies web plus proches de ma spécialisation SLAM.