

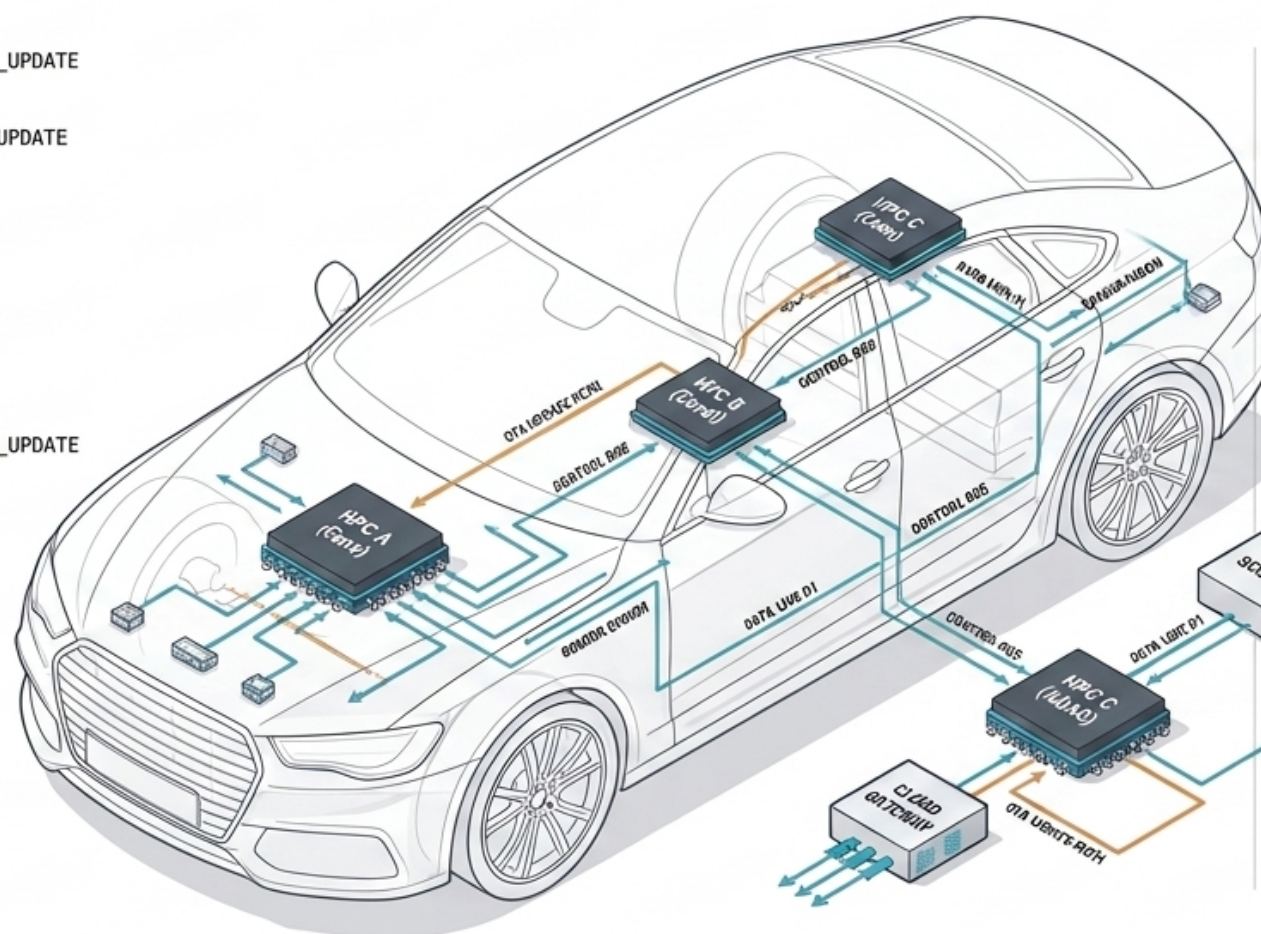
Maîtriser le Changement Logiciel en Vie Série

Processus de validation et déploiement pour le Software Defined Vehicle (SDV)

```

1 INIT_BOOT_SEQ >> START_V_SW_UPDATE
2 ...
3 VALIDATION_CHK: PASS
4 SYETREIP_READ >> START_V_S_UPDATE
5 VALIDATION_CHK: PASS
6 VALIDATION_CHK: PASS
7 OTA_PKG_RCV: SUCCESS
8 OTA_PK_ENPSN: SUCCESS
9 OTA_PKG_RCVY: SUCCESS
10 OTA_PKG_RCV: SUCCESS
11 OTA_PKG_RCV: SUCCESS
12 SYSTEM_STATUS: ONLINE
13 SYSTEM_STATUS: ONLINE
14 DEPLOY_VER_2.1.5
15 ...
16 INIT_BOOT_SEQ >> START_V_SW_UPDATE
17 VALIDATION_CHK: PASS
18 DEPLOY_VER_2.1.5
19 ...
20 OTA_PKG_2
21 OTA_PKG_RCV: SUCCESS
22 OTA_PKG_RCV: SUCCESS
23 DEPLOY_VER_2.1.5
24 DEPLOY_VER_2.1.5
25 ...
26 VAUTATION_CHK: PASS
27 VALIDATION_CHK: PASS
28 OTA_PKG_RCV: SUCCESS
29 SYSTEM_STATUS: ONLINE
30 SYSTEM_STATU: ONLINE
31 SYSTEM_STATUS: ONLINE
32 DEPLOY_VER_2.1.5
33 ...

```



```

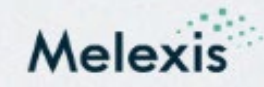
INIT_BOOT_SEQ >> START_V_SW_UPDATE
...
VALIDATION_CHK: PASS
VALIDIRIP_READ >> START_V_SW_UPDATE
VALIDATION_CHK: PASS
VALIDATION_CHK: PASS
OTA_PC_PCV: SUCCESS
OTA_PK_SNPEK: SUCCESS
OTA_PK_SRNSY: SUCCESS
...
OTA_HATER_CNK: PASS
...
SYSTEM_STATUS : RVALIDITE
SYSTEM_STATUS: PASS
DEPLOY_VER_2.1.5
...
OTA_PKG_RCV : 20S
DATA FORT!
...
PKG_2
SRV: OK
EEGR3IT: TRb
ITY_VER_2.1.5
DEPLOY_VER_2.1.5
...
VAUTASN_CHK: PASS
VATION_CHK: PASS
OTA_PKG_RCV: SUCCESS
SYSLOY_STATUS: ONLINE
...
SYSTEM_STATUS: ONLINE
DEPLOY_VER_2.1.5
...

```

Introduction – Participants



Svetlana
DICHEVA



Damien
CHARDAIRE



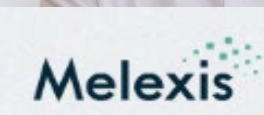
Patrice
KAHN



Guillaume
LUSSIER



Mickael
GRANDSIRE



Benjamin
COLLE



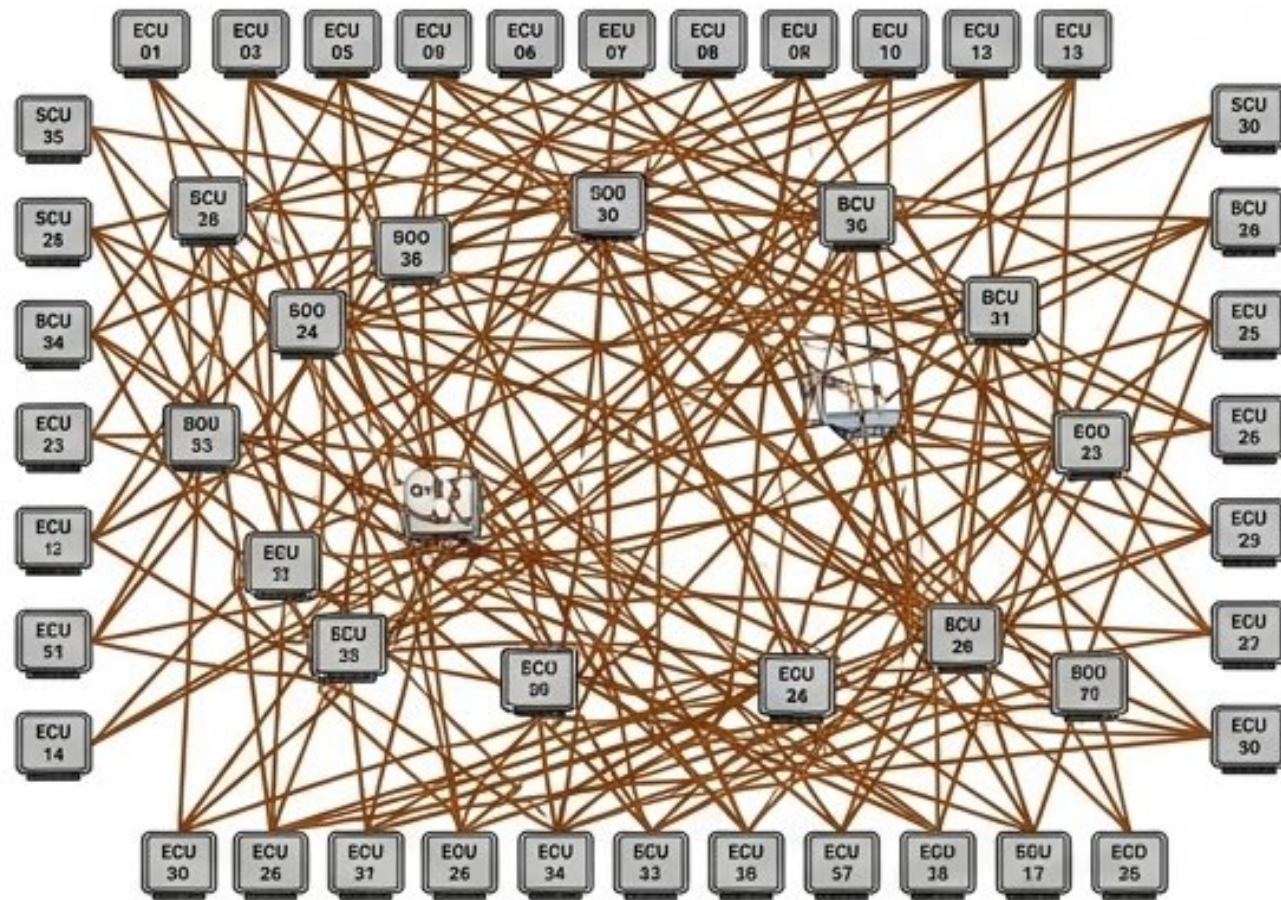
Benoit Bibie



Julien
CANCELIER

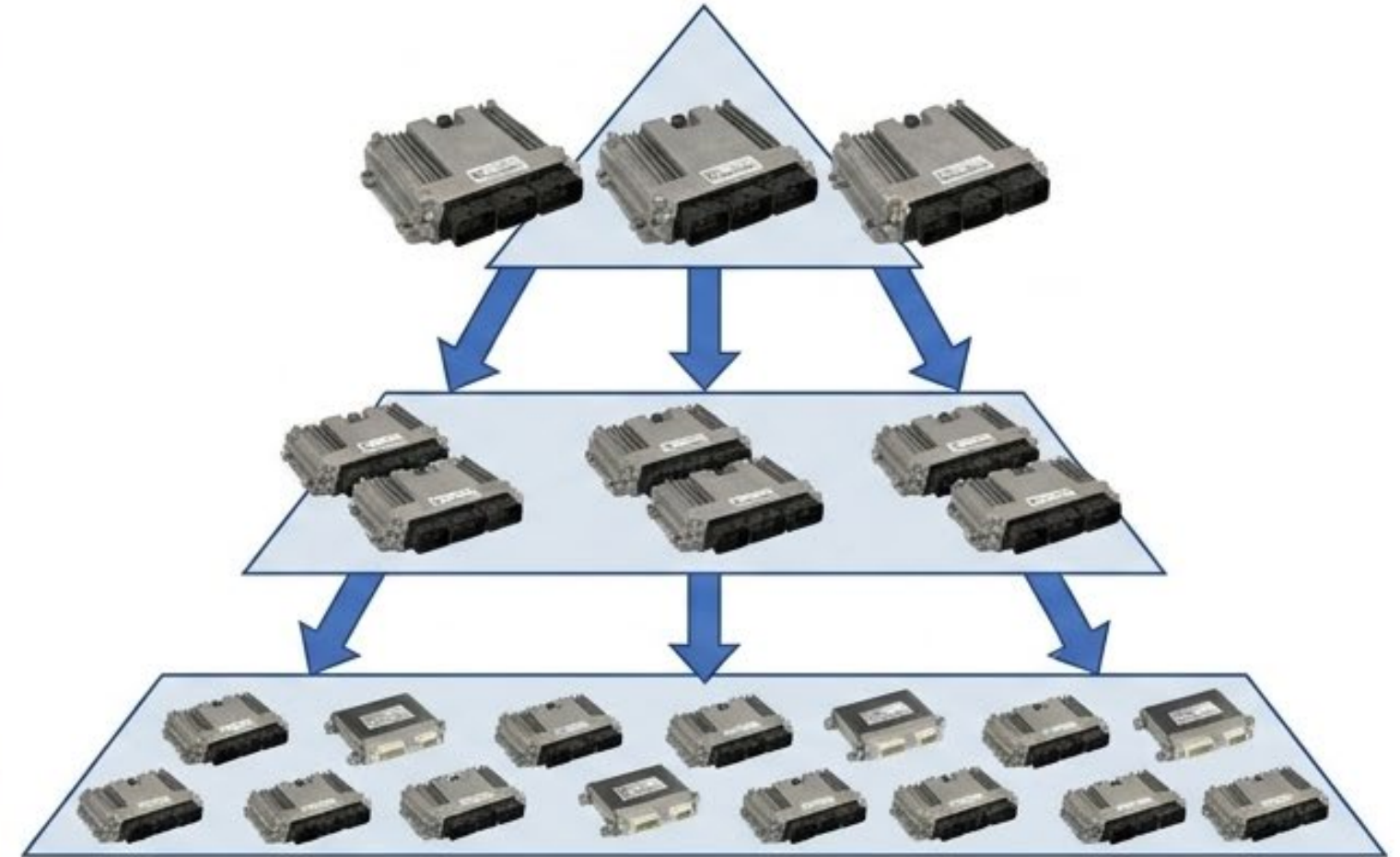
L'Évolution Incontournable vers le Software Defined Vehicle

Architecture Distribuée (HDV)



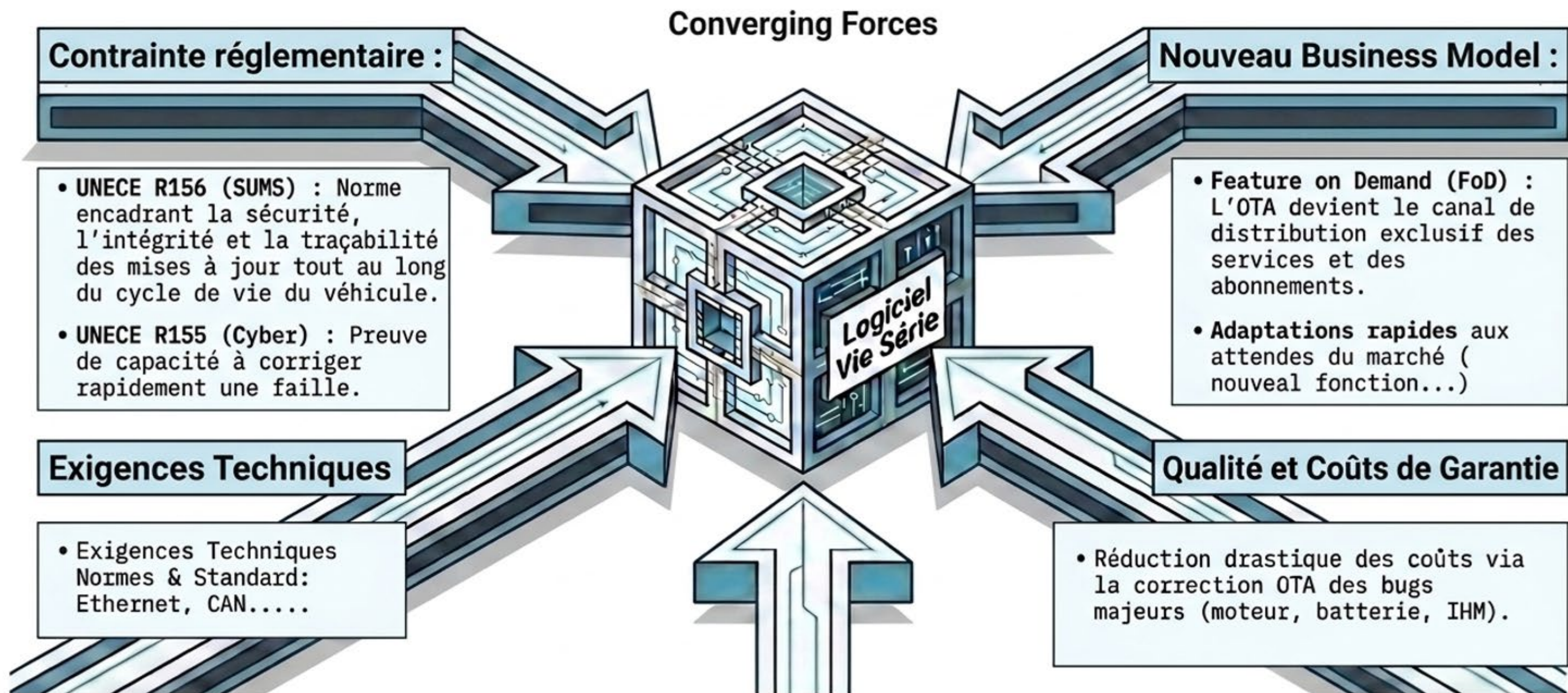
- 70 à 100 calculateurs (ECUs) isolés.
- 50 à 150 versions simultanées (Le cauchemar de gestion).
- **Dépendance au Niveau d'Intégration (I-Level)** pour la validation globale.

Architecture Zonale (SDV)



- Centralisation sur 3 à 5 calculateurs haute performance (HPCs).
- Explosion de la complexité logicielle (OS complets, hyperviseurs).
- Mise à jour continue : Le véhicule s'améliore dans le temps.

Nouveau contexte de développement



Typologie et Périmètre des Changements Logiciels

Urgent & Non-Planifié (< 1 mois)



- **Déclencheurs** : Corrections de bugs critiques (sécuritaire, fonctionnel), nouveaux risques liés à la cybersécurité.
- **Caractéristique** : Mode réactif immédiat.

Planifiable (Quelques semaines à 6 mois)

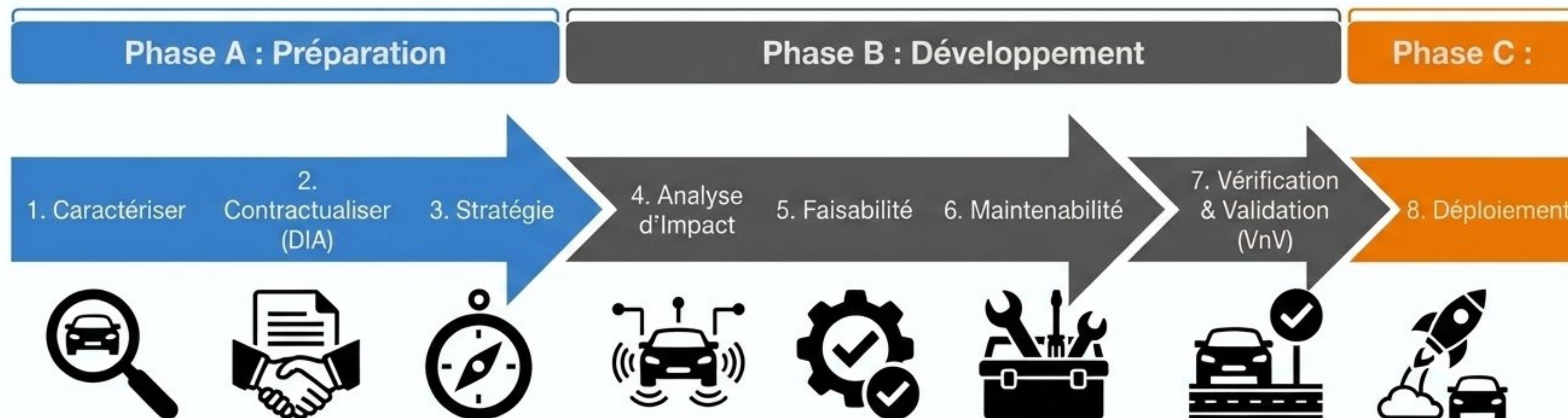


- **Déclencheurs** : Nouvelles fonctionnalités (Feature on Demand), PCN (Notification de changement de composant), traitement de l'obsolescence, amélioration continue.
- **Caractéristique** : Mode proactif, train de modifications regroupées.

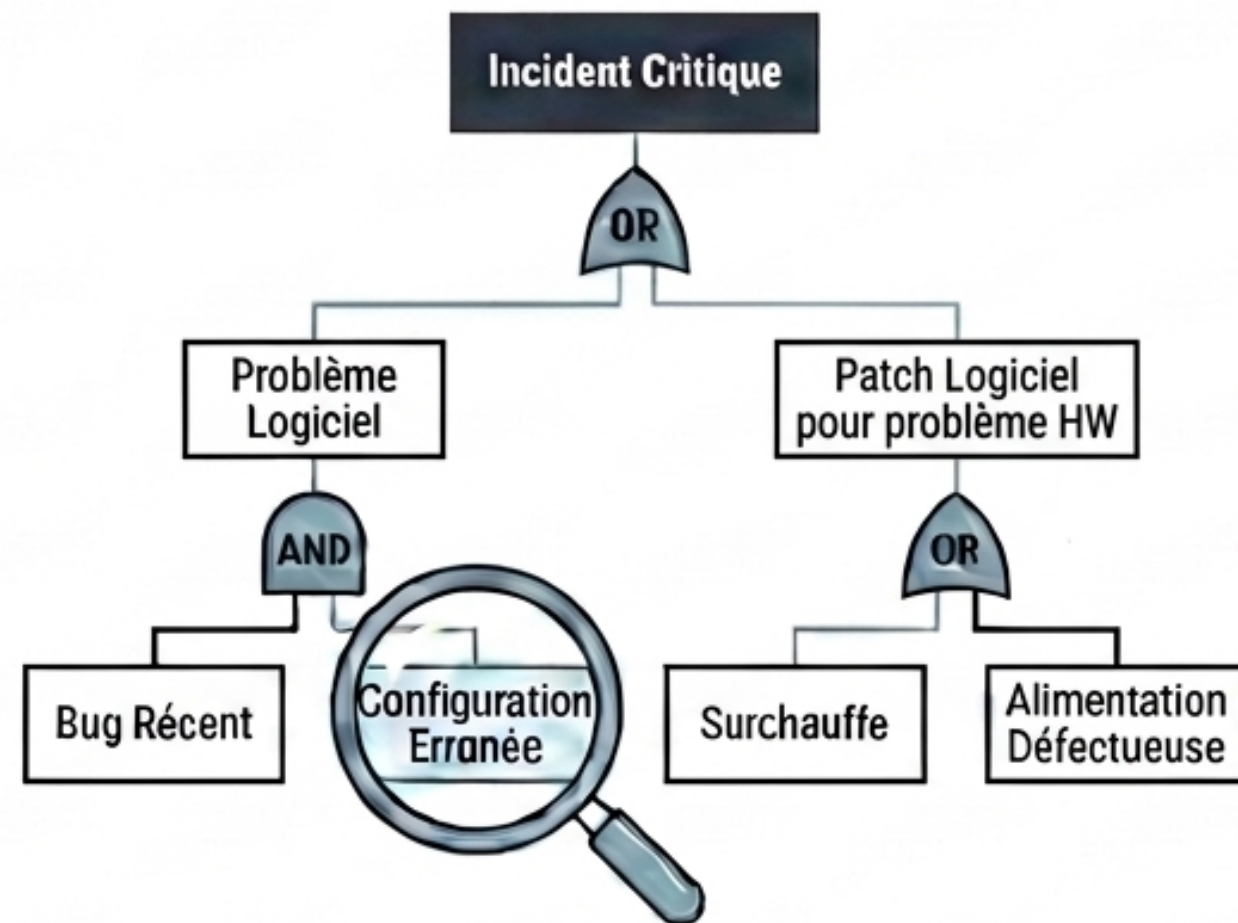


Alerte Criticité : Tout impact sur une fonction Safety ou réglementaire peut exiger une nouvelle homologation, redéfinissant intégralement le planning.

Le Moteur du Changement : 8 Étapes Critiques en Vie Série

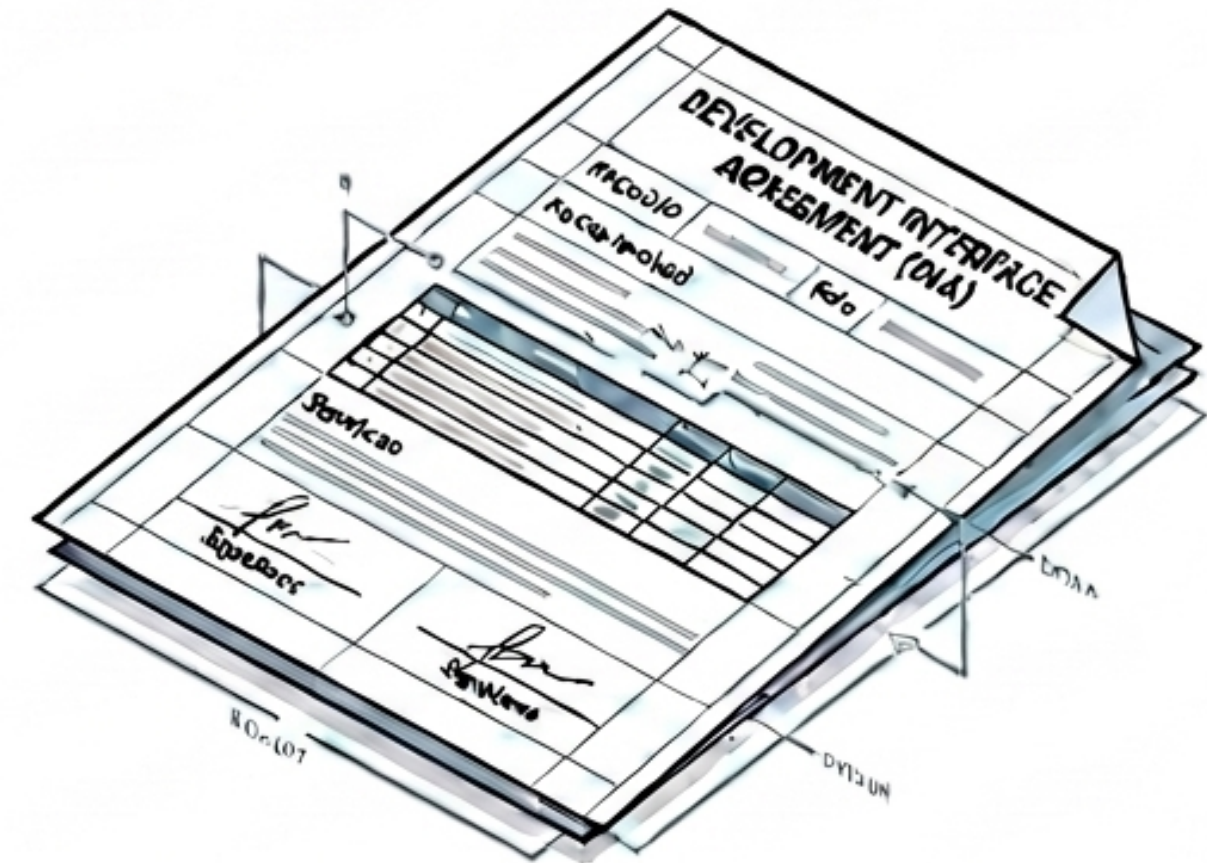


Cadrer l'Intervention avant d'Écrire le Code



Étape 1 : Caractériser et Geler le Périmètre

- **Le Risque** : L'over/under engineering causé par un manque de diagnostic.
- **L'Action** : Isoler la cause racine via FTA et 5-Why. Délimiter précisément les impacts sur le système.
- **La Règle** : Figer le périmètre pour éviter les effets de bord inattendus sur les performances temps réel (CPU/RAM).

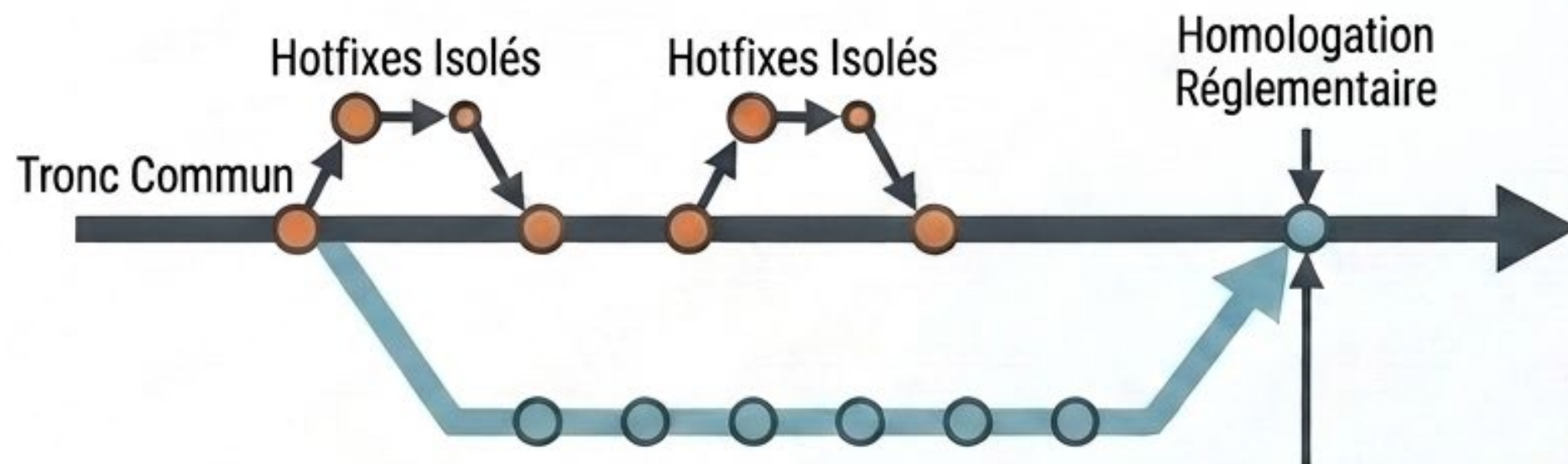


Étape 2 : Development Interface Agreement (DIA)

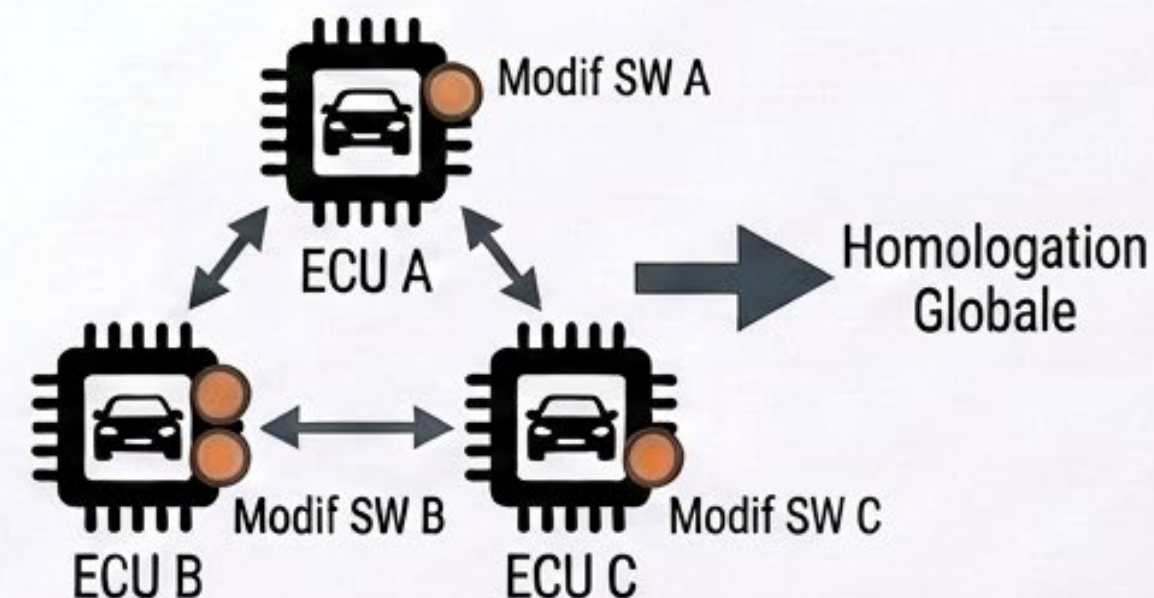
- **Le Principe** : Contractualiser dès le départ entre les différents intervenants du développement et de la validation: les rôles, les ressources et les modèles
- **L'Objectif** : Ne jamais avoir à négocier le Comment ou le Qui lors d'un changement d'urgence et critique.

Étape 3 : Orchestrer la Stratégie de Changement

Changement Mineur d'un ECU



Changement Majeur de ECUs Interconnectées



Optimisation des Branches

- Minimiser la fragmentation en limitant le nombre de branches maintenues.
- Regrouper les modifications non-urgentes pour diluer les coûts d'homologation et d'intégration système.

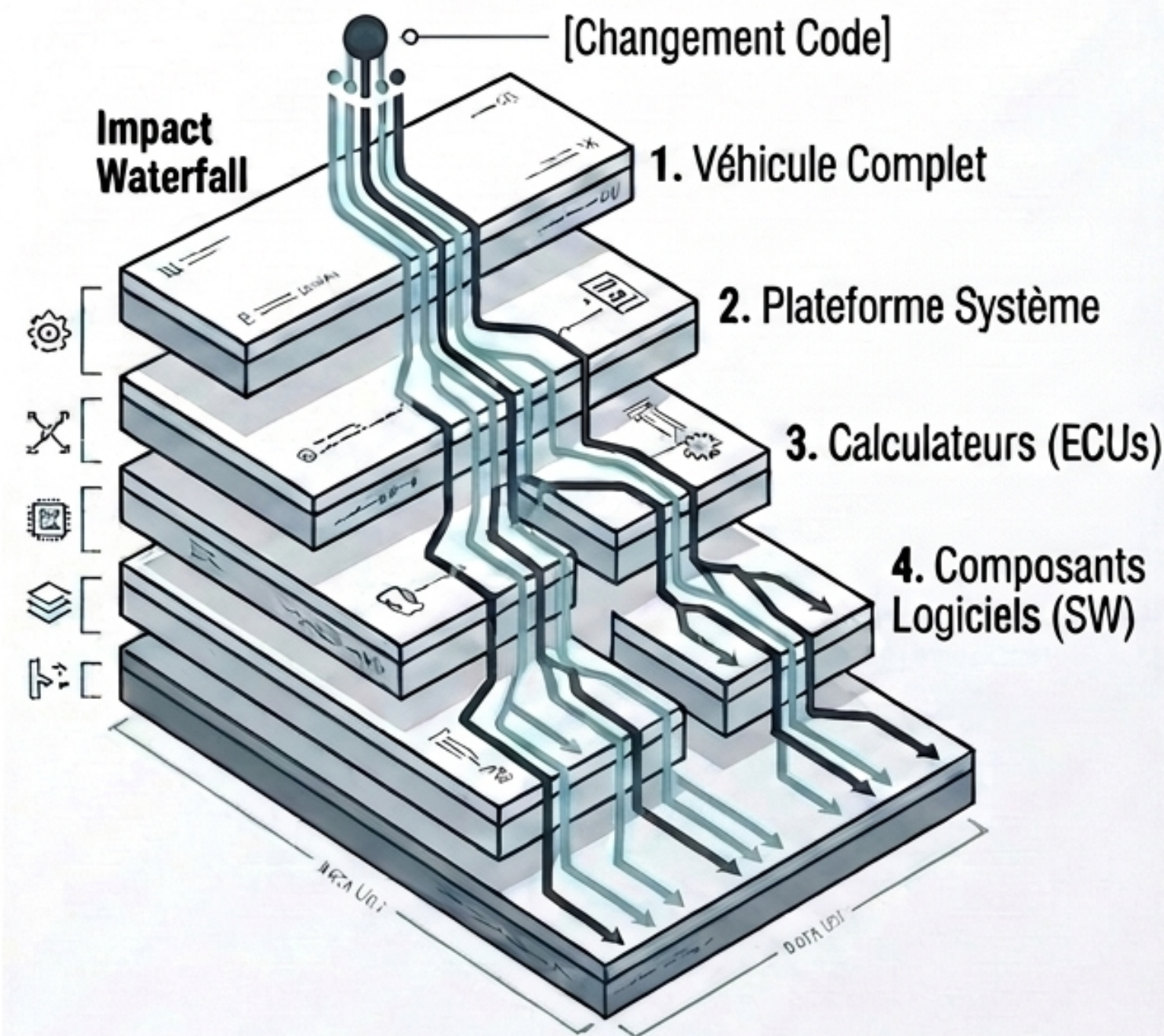
Gestion de la Fréquence

- Imposer une périodicité stricte (ex: mise à jour annuelle planifiée).
- **Objectif** : Limiter les perturbations et les désagréments pour le conducteur.

Le Filet de Sécurité (Rollback)

- Intégration d'une capacité système de retour automatique à la version N-1.
- Une solution vitale et temporaire pour sécuriser le véhicule en cas d'échec critique lors du déploiement.

Étape 4 : L'Effet Cascade : Cartographier l'Analyse d'Impact



Évaluer la Criticité de l'Impact

Le changement est défini comme **Majeur** s'il :

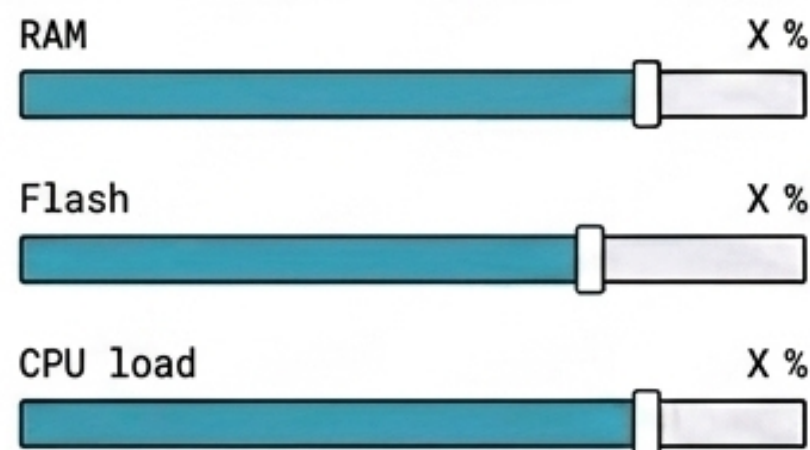
- Impacte plusieurs blocs distincts.
- Touche des fonctions transverses (Safety/Cyber).
- Nécessite la relance d'un cycle de développement global.

Les Trois Piliers d'Évaluation

- **Safety (ISO 26262)** : Analyse DFA (Dependant Failure Analysis) pour identifier et circonscrire les effets de bord inattendus.
- **Cybersécurité (ISO 24434)** : Évaluation stricte des nouvelles vulnérabilités potentiellement introduites.
- **Réglementaire** : Détermination définitive du besoin de re-homologation.

Etape 5&6 : Diagnostiquer la Faisabilité et la Maintenabilité

Faisabilité Technique (Limites Hardware)



- Vérification stricte de la compatibilité mémoire.
- Analyse de la charge CPU en temps réel via des outils de profiling.
- Intégration précoce des fournisseurs Tiers 2 & 3 pour éviter les angles morts techniques.

KPIs de Maintenabilité (Garantir l'Avenir)



Chaque objectif dépend de l'architecture, de l'application et du constructeur/ Tier1...

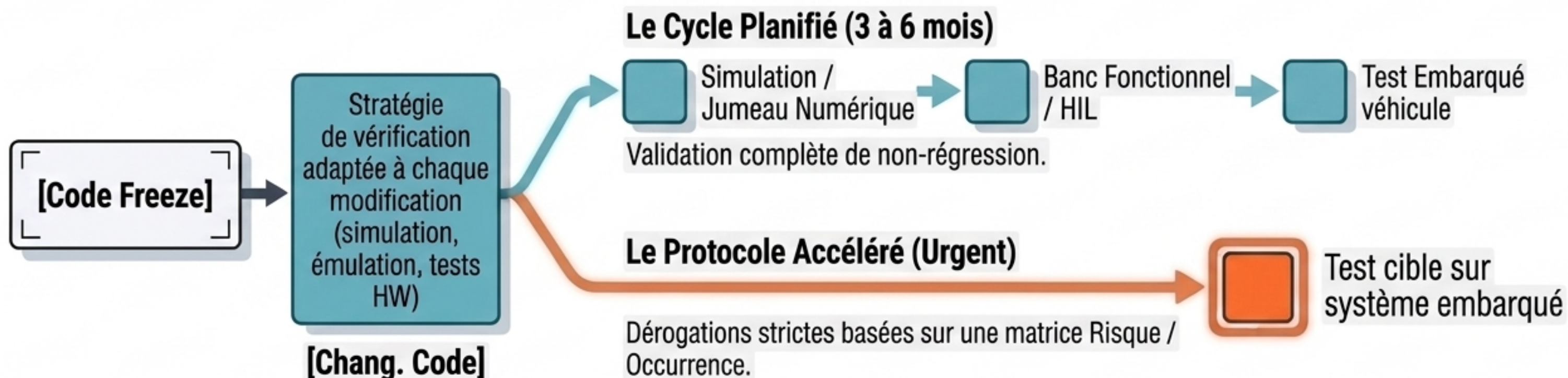
- **Complexité (McCabe)** : ≤ 10 par méthode (Garantit une lisibilité optimisée).
- **Dette Technique** : Ratio $\leq 5\%$ (Réduit les coûts de maintenance à long terme).
- **Maintainability Index (MI)** : Cible ≥ 85 (Très maintenable) / Alerte si ≤ 65 (Risque élevé).

Outillage Requis



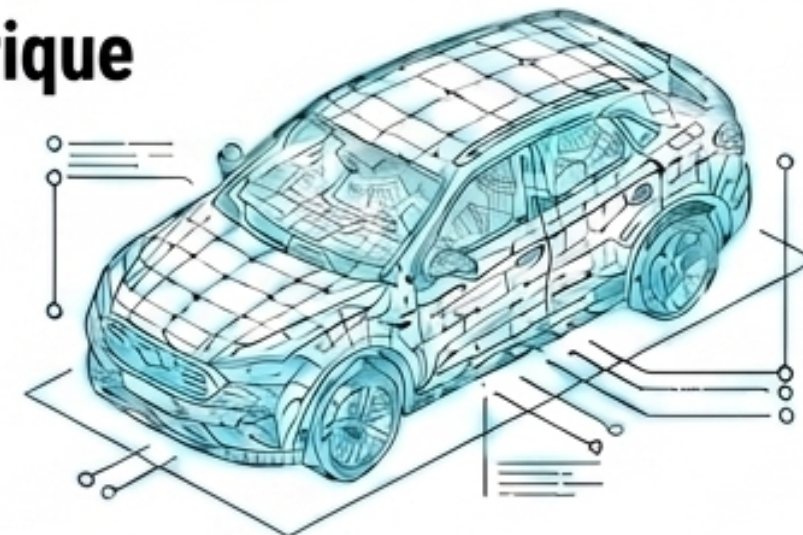
- Analyse automatisée et continue du code.
- Outils de référence : Static code analysis, Sound unsound / code coverage / opensource tools...
- Vérification stricte et impartiale des conventions de codage.
- CI/CD (Continuous Improvement Continuous Development & Deployment)

Etape 7 : Stratégie de Vérification et Validation (VnV) Adaptative

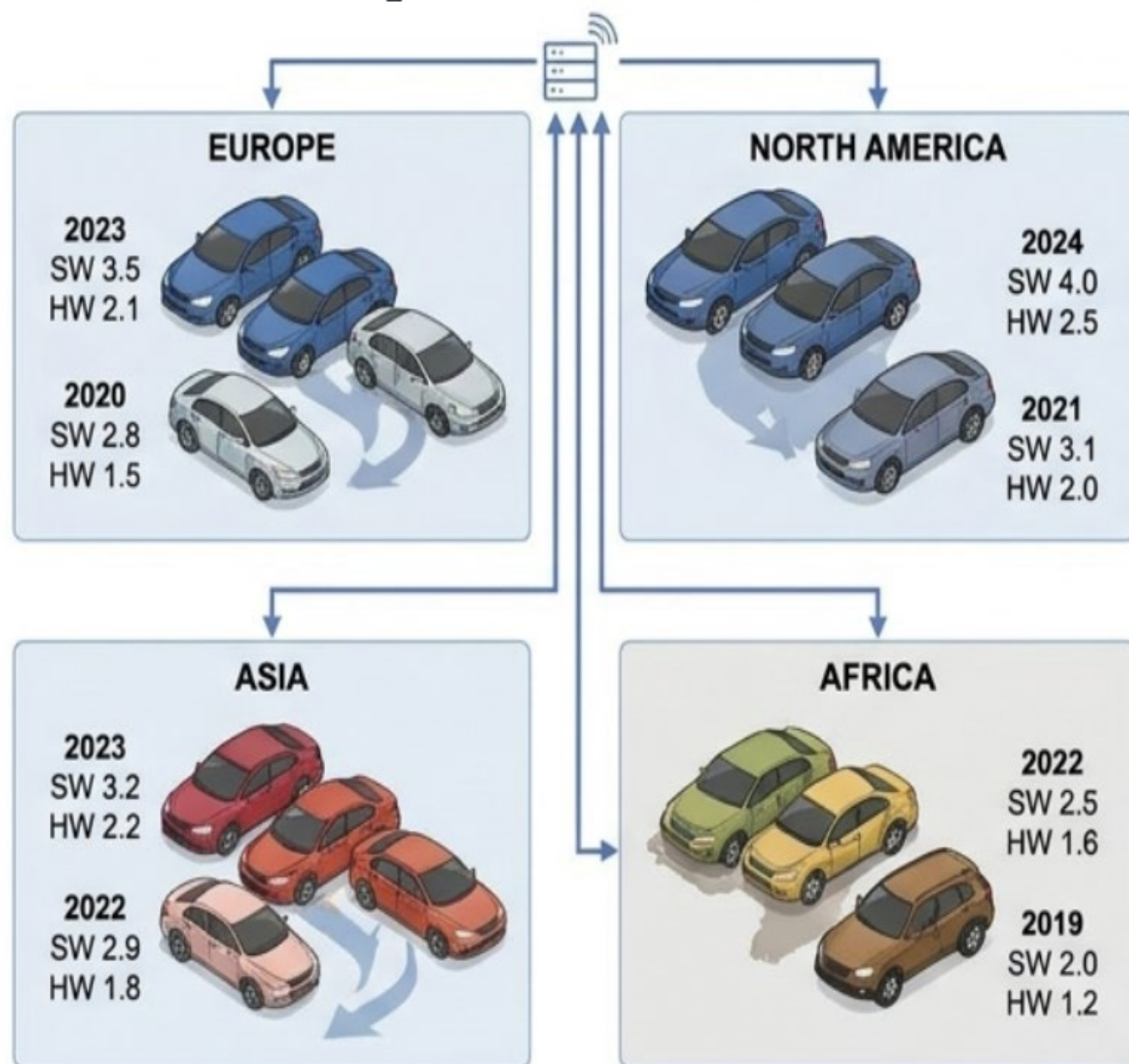


Le Catalyseur de Vitesse : Le Jumeau Numérique

La nécessité absolue de maintenir un jumeau numérique parfaitement synchronisé. Il permet la **parallélisation des tests** (simulation, émulation) pour compresser drastiquement les délais de validation avant tout déploiement matériel.



Déploiement Over-The-Air et Surveillance (QoS)



Gestion de l'Hétérogénéité des Flottes

- Télécharger la dernière configuration exacte du véhicule cible avant toute action.
- Gérer les variations matérielles (HW), les architectures réseaux et les législations spécifiques par pays.

Tolérance aux Pannes (Politique de 'Retry')

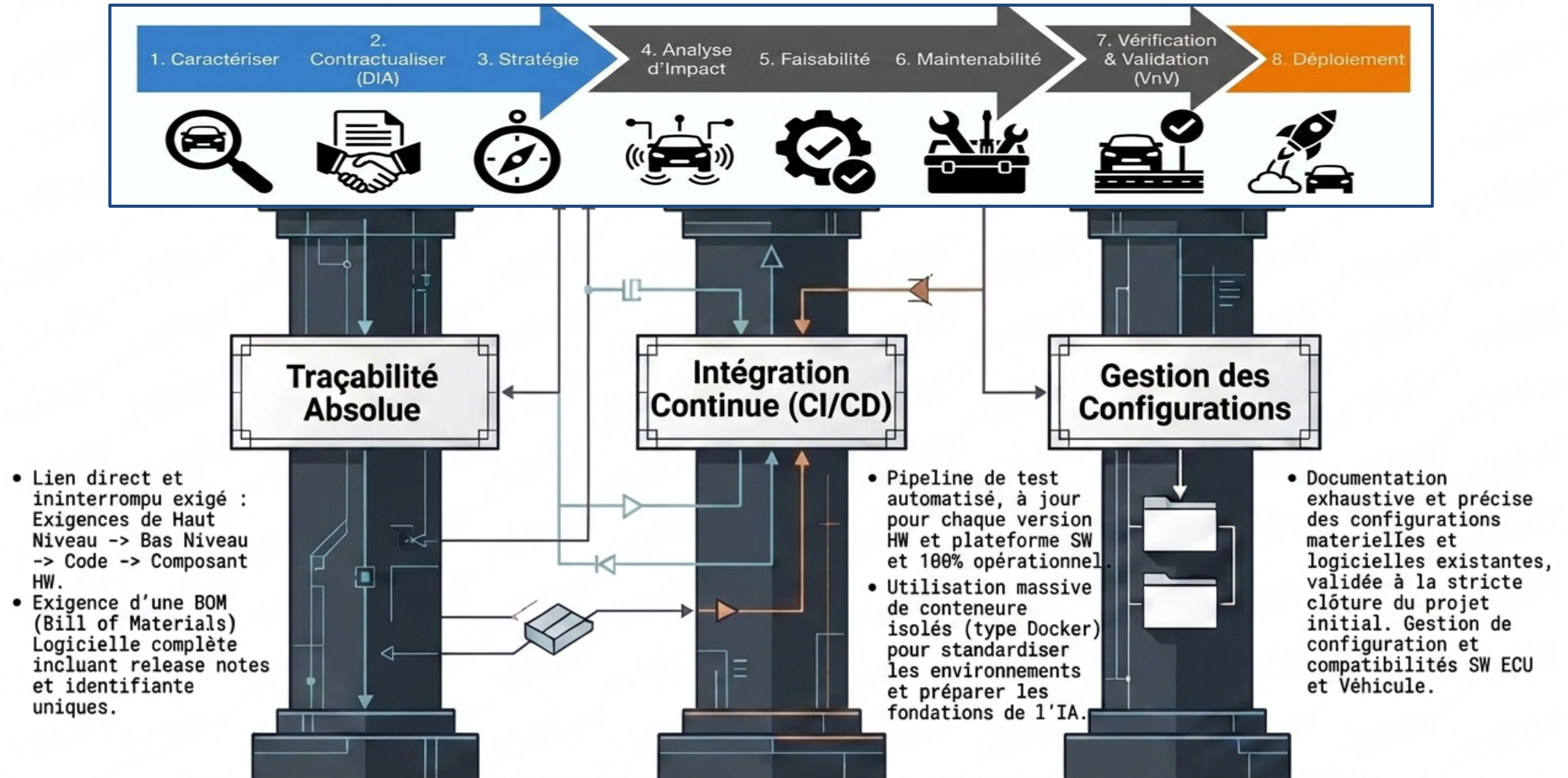
- Implémentation d'une architecture mémoire en 'Double Bank'.
- Permet le basculement instantané sur la version précédente en cas d'échec de mise à jour ou de perte réseau.

Monitoring Post-Déploiement (QoS)

- Surveillance continue de la cadence de déploiement (SLOW / FAST).
- Évaluation rigoureuse des taux de succès, remontée des défauts post-update, et gestion stricte du consentement utilisateur.

QUELS CRITERES DE REUSSITES ?

Les 3 facteurs clés de réussite



Le Changement de Paradigme Organisationnel

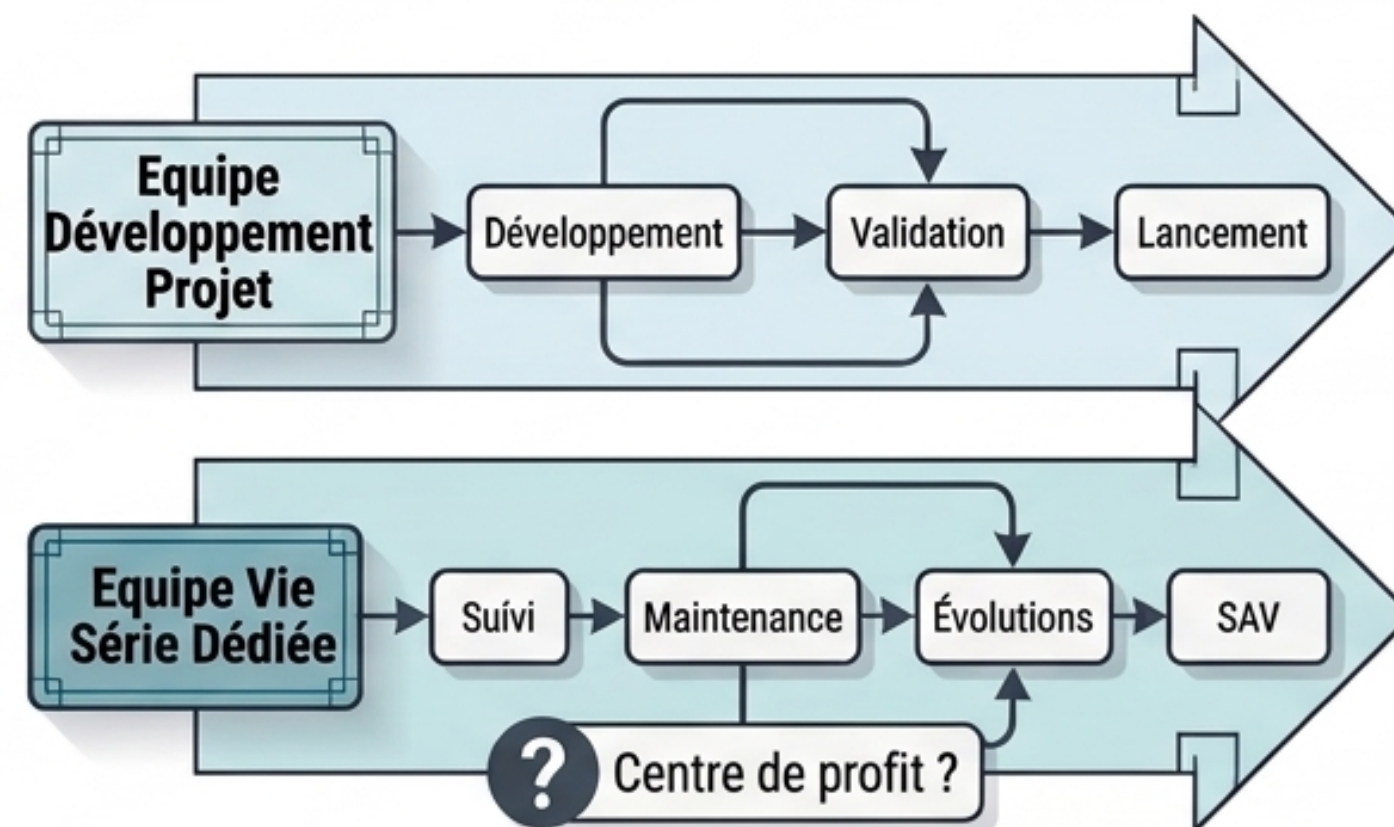
L'Ancien Modèle (Centre de Coût)



Le Piège de l'Équipe Projet :

- Les ingénieurs sont tiraillés entre le développement d'un nouveau véhicule et la correction urgente des bugs du modèle précédent.
- Les budgets et délais alloués aux corrections sont incompatibles avec le temps 'Projet', créant des conflits de priorisation massifs et un épuisement des ressources.

Le Nouveau Modèle (Centre de Profit)



L'Équipe Vie Série Dédinée :

- Création d'équipes autonomes, stables, expertes du processus de modification et séparées des développements initiaux.
- Des profils polyvalents ('T-Shape') agissant comme des 'urgentistes' du code plutôt que des chirurgiens ultra-spécialisés.
- Analogie : Le logiciel vie série devient un produit autonome, nécessitant ses propres usines et son propre 'SAV'.

Protocole de Gestion de Crise SW

Activation requise en cas de :



Risque de Rappel



Risque Sécuritaire (Safety)



Faible Cybersecurité Majeure

Accélérateurs de Réactivité



Checklists Obligatoires :
Déployées pour l'analyse
d'impact et la validation.

Objectif : Gain de temps,
guidage des activités et
prévention des erreurs
basiques.



Plans de Tests : Exécution
systématique d'un plan de
test minimum prédéfini.



Matrice RASI : Définition
stricte des rôles et
responsabilités pour chaque
contributeur, adaptée à
l'organisation en place.

Matrice d'Escalade



Vert

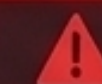


Jaune

Orange

Rouge

*L'analyse d'impact définit le niveau de criticité. Ce niveau
dicte l'allocation des ressources et les priorités de résolution.*



Protocole "Alerte Rouge"

**URGENCES ENTREPRISE (Safety,
Réglementaire, Cybersécurité)**

- **Validation Directe :** Approbation par
l'Autorité Qualité (Direction) obligatoire pour
supprimer les goulots d'étranglement
administratifs.
- **Organisation :** Basculement immédiat en
mode Task Force.
- **Allocation des Ressources :** Priorité absolue.
Dépassement des sprints Scrum avec
suspension autorisée des autres activités.
- **Rythme (Mode Agile) :** Alignement
permanent intensif (2 à 3 fois par jour avec
toutes les parties prenantes).

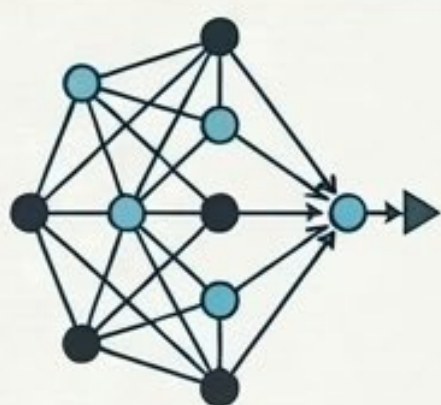
Gouvernance & Communication

Pilotes : 1 Chef de projet dédié
par partie prenante
(Carmaker, Tier X).

Stratégie : Communication
alignée en permanence vis-à-vis
de tout l'écosystème projet.

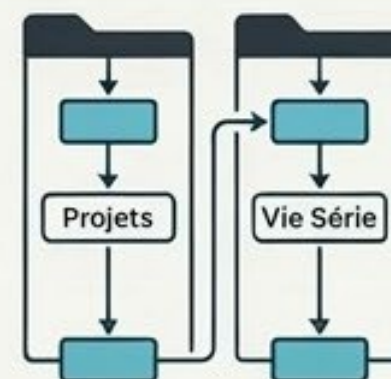
Reporting : Suivi strict,
transparent et régulier des
statuts d'avancement.

Synthèse : Maîtriser l'Ère du Software Defined Vehicle



1. Traçabilité et Rigueur Architecturale

Une modification d'apparence mineure peut invalider une homologation complète. L'analyse d'impact (l'effet cascade) est la clé de voûte de tout le processus.



2. Des Équipes Dédiées et Spécialisées

Séparer impérativement les développements Projets des urgences Vie Série. La gestion du changement continu est un métier à part entière.



3. Fournisseurs parties prenantes du changement

Ne jamais négocier pendant la tempête. Les accords d'interface de développement (DIA) doivent être figés en amont avec l'ensemble des fournisseurs.



4. Le Jumeau Numérique comme accélérateur

La simulation avancée et l'intégration continue sont les seules armes viables permettant de compresser des cycles de validation de 6 mois en quelques semaines.

Le logiciel n'est plus un composant figé à la sortie d'usine ; c'est un produit vivant qui redéfinit la valeur du véhicule à chaque mise à jour.