

## Présentation du socle Client/serveur « C/S »

### SOMMAIRE

<b>1. L'architecture « client Serveur » .....</b>	<b>2</b>
<b>1.1. L'architecture logique : Client Serveur.....</b>	<b>2</b>
1.1.1. Les différents clients .....	3
1.1.1.1. Clients Lourds.....	3
1.1.1.2. Clients Légers .....	3
1.1.2. Le dialogue entre le poste Client et le Serveur.....	4
1.1.3. Les programmes serveurs.....	4
1.1.4. Volumétrie des composants techniques .....	6
1.1.5. Le système de gestion de base de données.....	6
<b>2. Cycle et environnements de développement .....</b>	<b>7</b>

Service SI			Présentation du socle Client/serveur « C/S »		
Chrono :	2593	N°Publication :	1	Date de dernière modification	04/03/2025
				N° Page :	2

Ce document a comme objectif de présenter de façon succincte l'architecture dite « client serveur » (ou C/S )

# 1. L'architecture « client Serveur »

## 1.1. L'architecture logique : Client Serveur

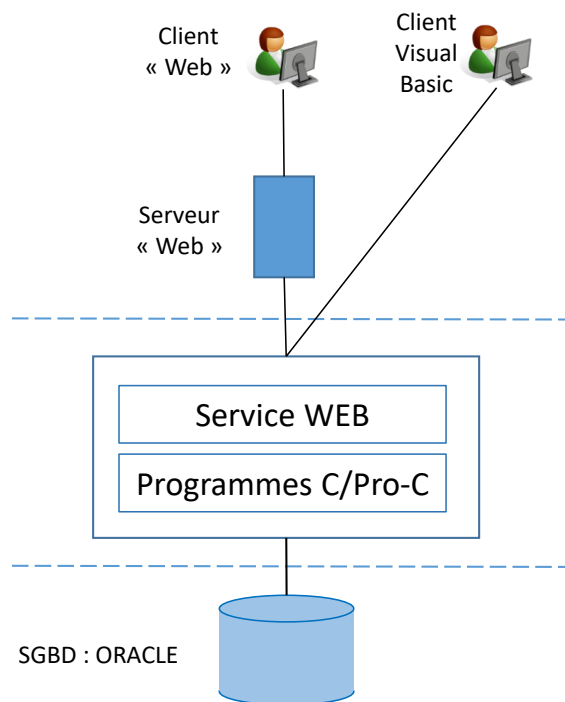
L'architecture repose sur un serveur d'applications qui centralise les informations et traite les demandes en provenance des postes ou machines clients. Les applications clientes n'ont donc qu'un accès indirect aux données, et ne servent que d'interface graphique pour les utilisateurs, l'ensemble des traitements étant laissé à la charge du serveur. La communication entre les postes, ou machines clientes, et le serveur, est assurée par l'utilisation de Services Web.

Les applications clientes, dont les besoins sont essentiellement graphiques, ont d'abord été développées en Visual Basic 6.0, et le sont dorénavant en Winforms (client dit lourd basé sur les technologies Microsoft .NET). Parallèlement, nous avons une partie dite « Web » développée en VB.NET sur la technologie ASP.NET. Une des particularités est d'ailleurs l'interconnexion entre les fenêtres Visual Basic et Winforms, ainsi qu'avec les applications « Web » dans certaines conditions.

Pour la partie serveur, tous les programmes sont écrits en PRO\*C (C + SQL).

Le système de gestion de base de données utilisé est ORACLE.

L'architecture de l'environnement de développement est une copie conforme de celle de production, afin de pouvoir tester les programmes développés dans des conditions proches de celle d'utilisation finale. Le service des Etudes dispose donc de son propre serveur d'applications et de ses propres données. Le transfert des programmes développés (ou maintenus) se fait uniquement lors de la phase de "Mise En Production".



**Figure 1 : Logique Client Serveur (n-tiers)**

Service SI			Présentation du socle Client/serveur « C/S »		
Chrono :	2593	N°Publication :	1	Date de dernière modification	04/03/2025
				N° Page :	3

La figure 2, ci dessus, met en évidence cette séparation stricte caractéristique des architectures n-tiers (à n=3). On observe bien les différents « tiers » :

- **Le tiers de présentation** : Composé des clients lourds (Visual Basic et Winforms), et clients légers (couple serveur Web et navigateur Internet).
- **Le tiers applicatif** : fonctionnel, il est composé de IIS, en tant que serveur d'applications, et de l'ensemble des programmes C/Pro\*C. Ce couple forme le serveur applicatif, et assure donc tous les traitements de la logique métier.
- **Le tiers de gestion des données** : Pris en charge par Oracle.

### 1.1.1. Les différents clients

L'architecture Client Serveur s'appuie sur 2 types de clients :

- Les clients lourds
- Les clients légers

L'application cliente effectue un minimum de traitements et de calculs. La fonctionnalité principale du client est la gestion des interfaces graphiques, c'est-à-dire :

- La récupération et le contrôle de la saisie utilisateur
- L'envoi des demandes de services
- La récupération et l'affichage de la réponse.

Cela signifie que le Client ne stocke aucune information et fait souvent appel aux services situés sur la partie Serveur. Ceci étant valable aussi bien pour les clients lourds (Visual Basic et Winforms) que pour les clients légers (clients Web).

#### 1.1.1.1. Clients Lourds

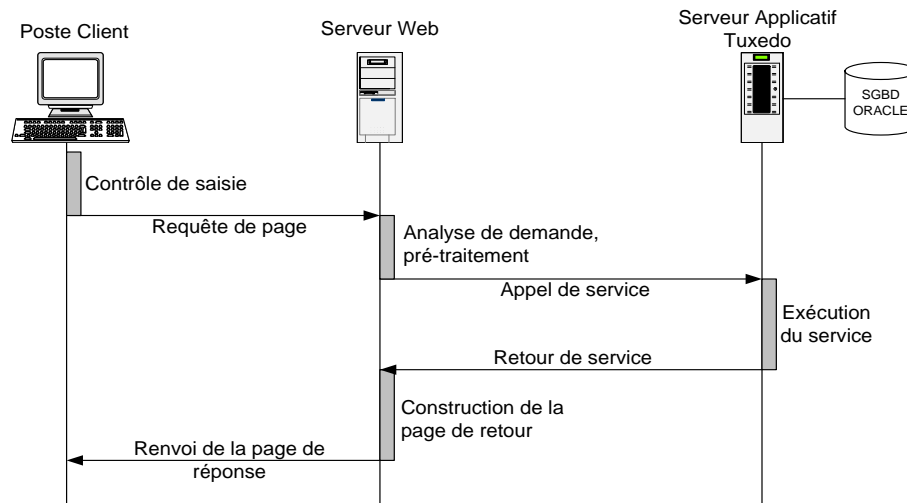
Pour ce qui est des clients dits « clients lourds », la partie cliente est composée de plusieurs exécutables (« projets » selon la dénomination Visual Basic ; on parle « d'ensembles fonctionnels » en ce qui concerne la technologie .NET). Chaque exécutable correspond à une application fonctionnelle telles que la gestion des quotas laitiers, la gestion financière, la gestion du courrier. Ces applications partagent des DLL (bibliothèques dynamiques) pour la gestion de certaines fonctionnalités communes techniques (telles que l'envoi des demandes de services et la réception des messages en réponse de ces services) ou métier (telles que la gestion des tiers). Une des caractéristiques de cette partie est l'absence de toute requête SQL. Le poste n'effectue que des demandes de services ce qui respecte le principe de séparation en couches, spécifique à ce type d'architecture.

#### 1.1.1.2. Clients Légers

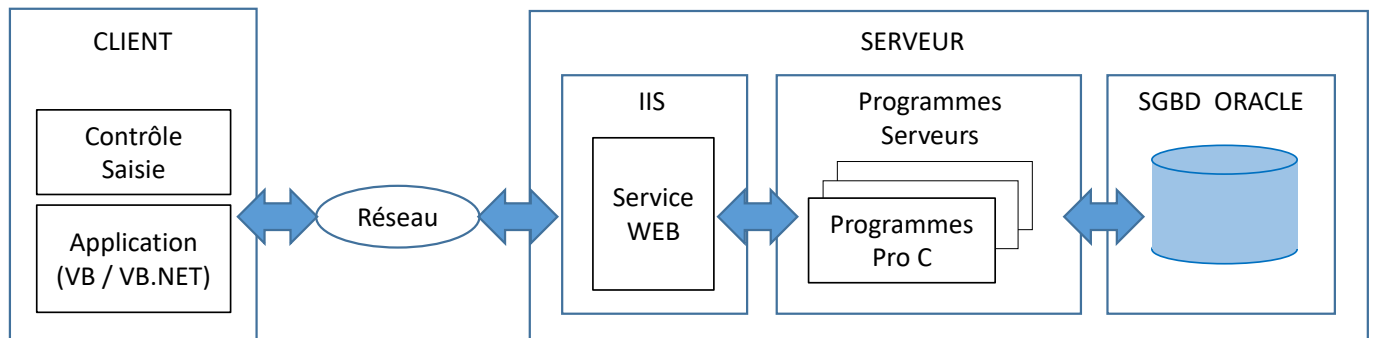
Pour les clients dits clients « légers », l'organisation est très proche de la partie « lourde ».

On retrouve la notion d'applications, mais celles-ci sont subdivisées en ensembles fonctionnels, qui regroupent des ensembles de pages à finalité fonctionnelle commune. Hormis la validation de saisie qui se fait sur le poste client (à l'aide de fonctions Javascript) tout traitement lié au client se fait sur le serveur Web (Intranet ou Extranet). Le dialogue client serveur peut s'illustrer comme suit :

Service SI		Présentation du socle Client/serveur « C/S »		
Chrono :	2593	N°Publication :	1	Date de dernière modification
				04/03/2025
			N° Page :	4



### 1.1.2. Le dialogue entre le poste Client et le Serveur



#### Dialogue entre un poste Client et le Serveur :

Par l'intermédiaire d'une application graphique VB ou Winforms (ou une page Web ASP.NET), les données saisies par l'utilisateur sont captées. Le « Client » va simplement effectuer une demande de service en passant en paramètre les données saisies par l'utilisateur. La demande du Client va alors être traitée par le Service Web qui va permettre de retrouver le service demandé.

Après l'exécution du service, les données en sortie sont retournées au Client. L'intérêt de cette architecture réside dans le fait que peu de données transitent sur le réseau. Il n'y a que des demandes ou des réponses de services. Au retour du service, le poste Client assure la mise à jour des applications graphiques.

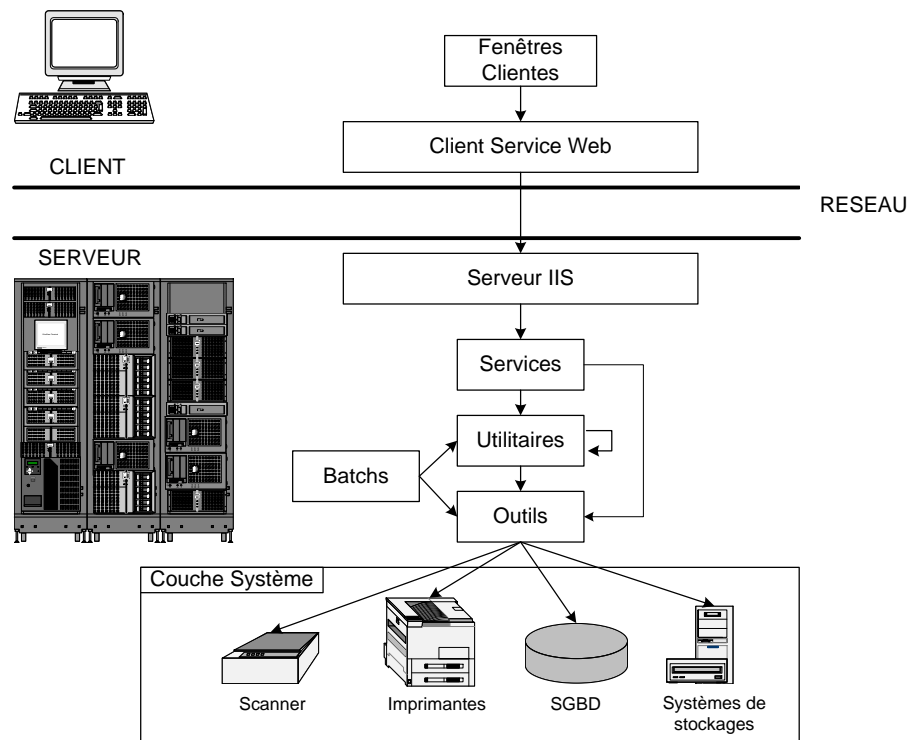
### 1.1.3. Les programmes serveurs

Le service Web permet d'invoquer des méthodes ou fonctions serveurs.

Ces fonctions sont contenues dans des fichiers bibliothèque de fonctions dynamiques (ou dll). Elles sont développées en Pro\*C, spécialisation du Langage C orienté SQL, permettant d'accéder simplement à ORACLE, et à ses fonctionnalités. Cependant, il a été convenu d'un découpage et d'une organisation des modules fonctionnels. Les seuls programmes invocables sont les *Services*, et les *Batches* (ayant la particularité supplémentaire d'être des exécutables).

Les autres composants ne sont appelables que selon la structure donnée plus en détail, ci après :

Service SI			Présentation du socle Client/serveur « C/S »		
Chrono :	2593	N°Publication :	1	Date de dernière modification	04/03/2025
				N° Page :	5



**Figure 2 : Modèle d'exécution et collaboration des programmes serveurs**

On trouve les éléments suivants :

- **Les Services** : Ils représentent l'unité de traitement fonctionnel invocable la plus élémentaire. Tout service peut faire appel à des utilitaires et à des outils.
- **Les Utilitaires** : Ils ne sont appelés que par les Batch, les Services ou d'autres Utilitaires. Ils exécutent des tâches spécifiques. Ils permettent également de réduire la taille des programmes qui les utilisent et d'avoir une architecture de programmes très modulaire.
- **Les Batches** : Ce sont des exécutables déclenchés par des événements à partir du serveur. Ils servent généralement dans le cadre d'extraction des données d'une des tables de la base de données sous Oracle, dans le cadre d'éditions de documents (souvent nocturnes) ou dans le cadre de traitements périodiques.
- **Les Outils** : Ils peuvent être appelés par n'importe quel type de programme. Ce sont les seuls programmes pouvant utiliser des fonctions internes (API, etc.) aux produits de l'architecture technique. Ils servent à abstraire le contexte technique. Ainsi, en cas de modification de l'environnement technique, seuls les Outils seront, normalement, à revoir. Ils servent, par exemple, à manipuler les fichiers, à quitter Oracle avec un Commit ou un Roll Back (ce qui permet de confirmer ou non les modifications apportées à la base de données), à changer une variable de type date pour pouvoir l'insérer dans la table Oracle, et bien d'autres fonctions aussi diverses.

<i>Service SI</i>		Présentation du socle Client/serveur « C/S »		
Chrono :	<b>2593</b>	N°Publication :	<b>1</b>	Date de dernière modification
				04/03/2025
			N° Page :	6

#### 1.1.4. Volumétrie des composants techniques

Chiffres arrêtés au 12/03/2013 :

Type	Nombre	Nombre de lignes de code
Service	5 034	3 008 958
Utilitaire	5 576	3 068 278
Batch	1 762	1 980 646
Outil	262	45 329
<b>Total</b>	<b>12 634</b>	<b>8 103 211</b>

Chiffres arrêtés au 01/01/2017 :

Type	Nombre	Nombre de lignes de code
Service	5 535	3 297 481
Utilitaire	6 026	3 275 326
Batch	1 913	2 185 202
Outil	265	44 743
<b>Total</b>	<b>13 739</b>	<b>8 802 752</b>

Chiffres arrêtés au 15/03/2021 :

Type	Nombre	Nombre de lignes de code
Service	5 734	3 422 219
Utilitaire	6 151	3 330 723
Batch	1 980	2 277 984
Outil	268	44 743
<b>Total</b>	<b>14 133</b>	<b>9 075 669</b>

Chiffres arrêtés au 15/03/2025 :

Type	Nombre	Nombre de lignes de code
Service	5 832	3 475 645
Utilitaire	6 224	3 354 379
Batch	1 966	2 302 790
Outil	290	44 844
<b>Total</b>	<b>14 312</b>	<b>9 177 658</b>

#### 1.1.5. Le système de gestion de base de données

Le système de gestion de bases de données choisi est Oracle (aujourd'hui en version 11g).

Dans les applications, Oracle n'est utilisé que comme stockeur d'informations. L'ensemble des informations est regroupé dans une seule et même base. C'est par le biais de nombreuses intégrités référentielles que la cohérence de toutes ces données est assurée.

Conformément à ce que nous avons indiqué plus haut, l'ensemble du dialogue programmes/SGBD est géré au travers du code SQL intégré à celui des programmes serveur.

Service SI			Présentation du socle Client/serveur « C/S »		
Chrono :	2593	N°Publication :	1	Date de dernière modification	04/03/2025
				N° Page :	7

## 2. Cycle et environnements de développement

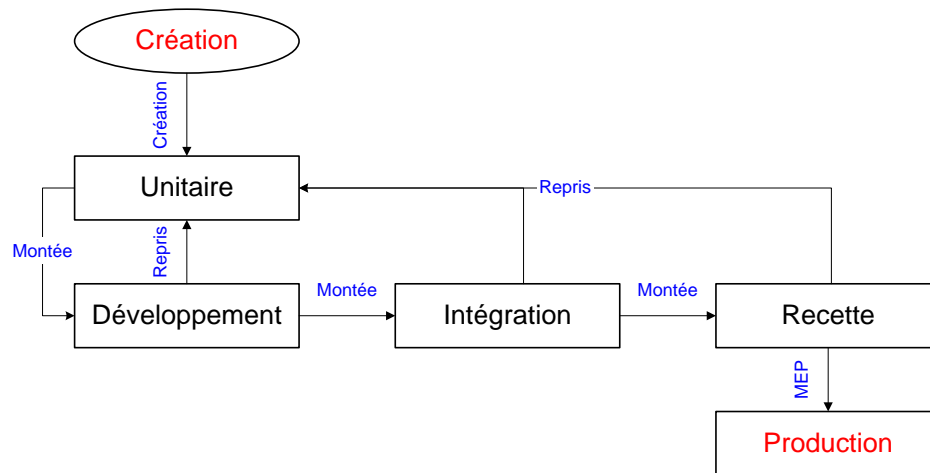
Les environnements de développements existants :

**Unitaire** : Environnement de conception et mise au point. A ce niveau, le code n'existe que sur la machine du développeur.

**Développement** : Premier environnement partagé par tous les développeurs. Il est utilisé pour réaliser les différents tests unitaires.

**Intégration** : Il permet aux analystes de valider une application avec tous les programmes qui la composent. Une fois l'application et ses programmes unitairement testés, les analystes s'occupent des tests d'intégration. Si les tests ne sont pas conformes, le programme redescend dans l'environnement unitaire, où il est repris, modifié, recompilé et re-testé. Une fois que le programme fonctionne, il remonte en Intégration où il subit le même examen, jusqu'à ce qu'il soit conforme et puisse monter à l'étape suivante, la Recette.

**Recette** : Environnement dédié aux tests utilisateurs. Les utilisateurs valident, à l'aide de jeux de tests qu'ils ont établis, le bon fonctionnement des applications, ainsi que l'adéquation avec les besoins exprimés en analyse. C'est le dernier niveau avant la mise en production.



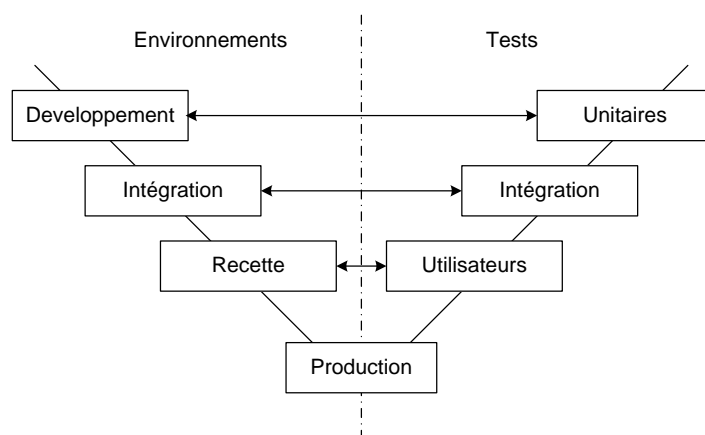
**Figure 3 : Cycle des environnements**

Chacun de ces environnements (sauf l'unitaire qui se sert de l'environnement de développement), est composé de l'ensemble des sources, fichiers compilés, exécutables et d'une base ORACLE spécifique permettant tous les tests, sans compromettre l'intégrité des données de production.

Tous les programmes sont référencés dans une table Oracle afin de pouvoir suivre leur développement. Cette table contient toutes les informations relatives au programme : nom, dates de création et de montée dans les environnements (développement, intégration, recette), environnement courant, type d'exécutable, nom du développeur, nom des documents d'analyse et de conception technique, nombre de lignes.

Toutes ces informations sont mises à jour automatiquement lorsqu'un programme est monté dans l'environnement supérieur, ou qu'il est repris pour une mise au point. Ceci est garanti par les outils de suivi du cycle de vie. Ces outils sont au nombre de trois : créer, version et repris. Ils garantissent la cohérence des informations de la table et sécurisent le développement en assurant la non concurrence des modifications (un seul développeur simultanément par fichier source). Ces outils ont été réalisés par la cellule technique, qui en assure donc la maintenance.

Service SI			Présentation du socle Client/serveur « C/S »		
Chrono :	2593	N°Publication :	1	Date de dernière modification	04/03/2025
				N° Page :	8



**Figure 4 : Correspondance entre Environnements & Tests**