

CENTRALE  
L Y O N

# GL - Frameworks

Approches pour la réutilisation  
Frameworks

BTD/MAI/FW 1

CENTRALE  
L Y O N

## Deux approches pour l'extensibilité

Bertrand DAVID : Méthodologies Avancées  
d'Informatisation

- L'approche **boite blanche** (transparente) pour obtenir l'extensibilité se base sur les concepts des langages à objets comme l'héritage et la liaison dynamique. Les fonctionnalités existantes sont réutilisées et étendues par l'héritage des classes de base du framework et la surcharge des méthodes prédéfinies.
- L'approche **boite noire** supporte l'extensibilité par la définition des interfaces des composants qui peuvent être enfichés dans le framework par la composition d'objets. Les fonctionnalités existantes sont réutilisées par la définition des composants qui sont conformes aux interfaces, l'intégration de ces composants dans le framework.

BTD/MAI/FW 2

CENTRALE L Y O N	<h2>Boite blanche et boite noire</h2>
Bertrand DAVID : Méthodologies Avancées d'Informatisation	<ul style="list-style-type: none"><li>● L'approche <b>boite blanche</b> nécessite une connaissance approfondie de la structure interne du framework. Les systèmes obtenus ainsi sont très intimement couplés aux détails de la hiérarchie d'héritage.</li><li>● L'approche <b>boite noire</b> est basée sur la composition et délégation plutôt que l'héritage. Il en résulte une réutilisation plus facile et une plus grande extensibilité. Ces frameworks sont par contre plus difficiles à développer car ils demandent aux développeurs une définition anticipée d'interfaces et de méthodes pour couvrir les cas d'utilisation potentiels.</li></ul>
BTD/MAI/FW	3

CENTRALE L Y O N	<h2>Object-oriented Application Framework (1)</h2>
Bertrand DAVID : Méthodologies Avancées d'Informatisation	<ul style="list-style-type: none"><li>● Les performances des machines et la puissance des réseaux ont considérablement augmentées dans la dernière décennie, tandis que la conception et la réalisation des logiciels complexes sont restées coûteuses et sources d'erreurs.</li><li>● Le coût et l'effort à fournir sont en grande partie dus à la découverte perpétuelle et la réinventions de concepts de base et de composants.</li><li>● En particulier, la hétérogénéité croissante des architectures matérielles, la diversité des systèmes d'exploitation et de plates-formes de communication rendent difficile la construction d'applications correctes, portables, performantes et peu coûteuses à partir de rien.</li></ul>
BTD/MAI/FW	4

CENTRALE L Y O N	<h2>Object-oriented Application Framework (2)</h2>
Bertrand DAVID : Méthodologies Avancées d'Informatisation	<ul style="list-style-type: none"><li>● <b>Object-oriented application frameworks</b> constituent une approche prometteuse pour proposer des conceptions et des implémentations de logiciels dans le but de réduire le coût et d'augmenter la qualité du logiciel.</li><li>● Le framework (cadre) est une application semi-finie qui peut être spécialisée pour produire une application adaptée.</li><li>● En contraste avec l'approche objet classique, dans laquelle la réutilisation était basée sur une bibliothèque de classes, les cadres (frameworks) visent des entités applicatives particulières (comme la téléphonie mobile) et des domaines applicatifs (IHM embarquée du pilote).</li></ul>
BTD/MAI/FW	5

CENTRALE L Y O N	<h2>Object-oriented Application Framework (3)</h2>
Bertrand DAVID : Méthodologies Avancées d'Informatisation	<ul style="list-style-type: none"><li>● Les frameworks comme MacApp, AT++, Interviews, ACE, Microsoft MFC et DCOM, RMI de JavaSoft et les implémentations de CORBA jouent un rôle croissant dans le développement actuel des logiciels.</li><li>● Le gain initial des frameworks provient de la modularité, la réutilisabilité, l'extensibilité et de l'inversion du contrôle qu'ils fournissent aux développeurs.</li><li>● La modularité est accrue par l'encapsulation des détails d'implémentation derrière des interfaces stables.</li></ul>
BTD/MAI/FW	6

CENTRALE L Y O N	<h2>Object-oriented Application Framework (4)</h2>
Bertrand DAVID : Méthodologies Avancées d'Informatisation	<ul style="list-style-type: none"><li>● La modularité du framework aide à augmenter la qualité du logiciel par la localisation de l'impact des changements de la conception et de l'implémentation ce qui réduit l'effort demandé pour comprendre et maintenir un logiciel existant.</li><li>● Les interfaces stables proposées par des frameworks augmentent la réutilisation par la définition de composants génériques qui peuvent être réappliquées pour créer de nouvelles applications.</li><li>● La réutilisation du framework sert de levier à l'effort fourni lors de l'acquisition des connaissances en évitant la nouvelle conception et la revalidation des solutions de base pour des besoins récurants.</li></ul>
BTD/MAI/FW	7

CENTRALE L Y O N	<h2>Object-oriented Application Framework (5)</h2>
Bertrand DAVID : Méthodologies Avancées d'Informatisation	<ul style="list-style-type: none"><li>● La réutilisation des composants du framework conduit à l'augmentation substantielle de la productivité et à une meilleure qualité, performance, fiabilité et interopérabilité.</li><li>● L'extensibilité du framework est obtenue grâce aux méthodes qui peuvent être spécialisées sans remettre en cause la stabilité des interfaces. Ces méthodes découplent la stabilité des interfaces de la variation nécessaire lors de l'instanciation de l'application dans un contexte particulier.</li><li>● L'extensibilité du framework est essentielle pour permettre adaptation de nouveaux services et attributs.</li></ul>
BTD/MAI/FW	8

CENTRALE L Y O N	<h2>Object-oriented Application Framework (6)</h2>
Bertrand DAVID : Méthodologies Avancées d'Informatisation	<ul style="list-style-type: none"><li>● L'architecture d'exécution du framework est caractérisée par l'inversion du contrôle. Cette architecture permet aux éléments canoniques de l'application d'être spécialisées par des objets gestionnaires d'événements qui sont évoqués par le mécanisme du framework qui assure la distribution réactive.</li><li>● Quand un événement apparaît, le distributeur du framework réagit en évoquant les méthodes des objets gestionnaires qui effectuent des traitements spécifiques. L'inversion du contrôle permet au framework (et non à chaque application) de déterminer quel est l'ensemble de méthodes spécifiques qui doivent être activées en réponse aux événements externes (comme des messages de fenêtres venant des utilisateurs).</li></ul>
BTD/MAI/FW	9

CENTRALE L Y O N	<h2>Des frameworks les plus courants</h2>
Bertrand DAVID : Méthodologies Avancées d'Informatisation	<ul style="list-style-type: none"><li>● Interfaces graphiques</li><li>● ORB (object request broker) qui facilite la communication entre objets locaux et distants.</li></ul>
BTD/MAI/FW	10

CENTRALE L Y O N	<h2>Typologie des frameworks</h2>
Bertrand DAVID : Méthodologies Avancées d'Informatisation	<ul style="list-style-type: none"> <li>• <b>Framework d'infrastructure des systèmes</b> pour obtenir des systèmes portables et performants comme des systèmes d'exploitation et de communication, des framework d'interfaces utilisateurs et outils de traitement de langages. Ces frameworks sont principalement utilisés de façon interne dans les équipes de développement et ne sont pas diffusés vers les clients.</li> <li>• <b>Frameworks de middleware</b> sont utilisés généralement pour bâtir des applications distribuées en intégrant des composants. Ils augmentent la modularité, la réutilisation et par là la fiabilité d'infrastructure à comportement semblables dans différents environnements distribués.</li> <li>• <b>Frameworks d'applications</b> pour les entreprises couvrent un ensemble croissant de domaines d'applications comme des finances, la productique, les télécommunications,... En comparaison avec les deux autres types de frameworks ces derniers sont à la fois beaucoup plus coûteux à développer ou à acquérir, mais génèrent également un retour sur investissement car ils supportent le développement direct d'applications finalisées pour les clients. Ceci contraste avec les deux premiers qui ne concernent qu'indirectement les clients finaux.</li> </ul>
BTD/MAI/FW	11

CENTRALE L Y O N	<h2>Avantages et inconvénients (1)</h2>
Bertrand DAVID : Méthodologies Avancées d'Informatisation	<ul style="list-style-type: none"> <li>• Les frameworks en conjonction avec des patterns et des bibliothèques de classes peuvent augmenter de façon significative la qualité et réduire l'effort de développement.</li> <li>• Pour cela il faut savoir que : Si la difficulté de développement d'un logiciel complexe, celle du framework correspondant le sera également.</li> <li>• La connaissance de développement reste encore et toujours dans les têtes des experts. L'apprentissage de l'utilisation d'un framework est longue (6-12 mois) mais l'investissement devrait s'amortir sur plusieurs projets.</li> </ul>
BTD/MAI/FW	12

CENTRALE L Y O N	<h2>Avantages et inconvénients (2)</h2>
Bertrand DAVID : Méthodologies Avancées d'Informatisation	<ul style="list-style-type: none"><li>● Le développement d'applications devrait progressivement intégrer plusieurs frameworks (IHM, Communication,...). Malheureusement des frameworks ont été principalement conçu dans l'optique d'évolution interne que dans celle d'intégration avec d'autres frameworks.</li><li>● Evolution d'applications peut conduire à l'évolution du framework.</li><li>● Maintenance du framework : fonctionnelle, non-fonctionnelle, généralisation, est une activité coûteuse.</li></ul>
BTD/MAI/FW	13

CENTRALE L Y O N	<h2>Positionnement (1)</h2>
Bertrand DAVID : Méthodologies Avancées d'Informatisation	<ul style="list-style-type: none"><li>● Un <b>framework</b> est une architecture réutilisable de l'ensemble ou d'une partie d'un système représenté par un ensemble de classes abstraites et la façon dont leurs instances interagissent.</li><li>● Un <b>framework</b> est un squelette d'une application qui peut être adapté par le développeur de l'application.</li><li>● Un <b>pattern</b> décrit un problème à résoudre, une solution et le contexte dans lequel il travaille.</li><li>● Les <b>patterns</b> ont pour but de décrire des solutions récurrentes.</li><li>● Un <b>framework</b> contient en général plusieurs <b>patterns</b>. Les patterns sont plus abstraits que les frameworks.</li></ul>
BTD/MAI/FW	14

CENTRALE L Y O N	<h2>Positionnement (2)</h2>
Bertrand DAVID : Méthodologies Avancées d'Informatisation	<ul style="list-style-type: none"><li>● Des patterns de conception sont des micro-éléments architecturaux d'un framework.</li><li>● Les frameworks sont au centre des techniques de réutilisation. Ils sont plus abstraits et plus flexibles que des composants, mais plus concrets et plus faciles à réutiliser que les patterns de conception.</li><li>● Les frameworks réutilisent à la fois la conception et le code.</li><li>● Les patterns sont illustrés par des programmes, mais les frameworks sont les programmes.</li><li>●</li></ul>
BTD/MAI/FW	15