

CENTRALE
L Y O N

UML et le Développement de Logiciels

Des facteurs aux critères

BTD/UML-DL 1

CENTRALE
L Y O N

Bertrand DAVID :
UML et le Développement de Logiciels

Les facteurs : caractéristiques d'utilisation

- liées à l'environnement d'exploitation
 - **Confidentialité**: aptitude d'un logiciel à être protégé contre tout accès par des personnes non autorisées.
 - **Couplabilité**: aptitude d'un logiciel à être couplé à un autre logiciel.
 - **Efficacité**: aptitude d'un logiciel à minimiser l'utilisation des ressources disponibles.
 - **Maniabilité**: aptitude d'un logiciel à être convivial et facile d'emploi pour l'utilisateur auquel il est destiné.
 - **Robustesse**: aptitude d'un logiciel à conserver un comportement conforme aux besoins exprimés en présence d'événements non souhaités ou non prévus.

BTD/UML-DL 2

CENTRALE L Y O N	<h2>Les facteurs : caractéristiques d'utilisation</h2>
Bertrand DAVID : UML et le Développement de Logiciels	<ul style="list-style-type: none">● liées à l'environnement de maintenance et de suivi<ul style="list-style-type: none">→ Maintenabilité: aptitude d'un logiciel à faciliter la localisation et correction des erreurs résiduelles.→ Adaptabilité: aptitude d'un logiciel à faciliter la suppression ou l'évolution de fonctionnalités existantes ou l'adjonction de nouvelles fonctionnalités.→ Portabilité: aptitude d'un logiciel à minimiser les répercussions d'un changement d'environnement logiciel et matériel.→ Testabilité: aptitude d'un logiciel à permettre la vérification de son fonctionnement.→ Observabilité: aptitude d'un logiciel à fournir des informations sur son fonctionnement.→ Réutilisabilité: aptitude d'un logiciel à être intégré en partie ou en entier dans d'autres logiciels.
BTD/UML-DL	3

CENTRALE L Y O N	<h2>Critères</h2>
Bertrand DAVID : UML et le Développement de Logiciels	<ul style="list-style-type: none">● Les facteurs traduisent une vision externe (son utilisation). Pour caractériser de façon interne on utilise des critères:<ul style="list-style-type: none">→ Banalité d'emploi: Indépendance par rapport à une application.→ Communicabilité: Facilité de dialogue entre l'homme et la machine.→ Efficacité mémoire: Consommation minimale de place mémoire.→ Efficacité des périphériques: Vitesse et capacité optimales des périphériques.→ Efficacité temps d'exécution: Consommation minimale de temps machine.→ Expansibilité: Possibilité d'accroissement des zones de données et de la taille des programmes.→ Exploitabilité: Facilité de mise en œuvre et d'utilisation.
BTD/UML-DL	4

CENTRALE L Y O N	<h2>Critères</h2>
Bertrand DAVID : UML et le Développement de Logiciels	<ul style="list-style-type: none"> → Facilité d'apprentissage: Facilité d'apprentissage des procédures et des opérations nécessaires au fonctionnement d'un programme. → Historique des accès: Mémorisation des accès effectués à un programme, une zone de données ou un fichier. → Indépendance/ environnement logiciel: Absence de liens structurels avec l'environnement logiciel. → Indépendance/ environnement matériel: Absence de liens structurels avec l'environnement matériel. → Lisibilité: Possibilité de compréhension d'un document ou d'un code par sa simple lecture. → Modularité: Décomposition d'un logiciel en éléments de taille réduite. → Observabilité: Facilité de localisation des non-conformités. → Précision: Exactitude des résultats obtenus.
BTD/UML-DL	5

CENTRALE L Y O N	<h2>Critères</h2>
Bertrand DAVID : UML et le Développement de Logiciels	<ul style="list-style-type: none"> → Protection du code et des données (en exploitation): Protection contre des accès par des personnes non autorisées, le logiciel étant en exploitation. → Protection du code et des données (hors exploitation): Protection contre des accès par des personnes non autorisées en dehors d' exploitation. → Simplicité: Facilité de compréhension liée à l'absence d'éléments superflus. → Standardisation des données: Compatibilité des données avec des standards de représentation. → Standardisation des interfaces: Compatibilité des interfaces avec des standards externes. → Tolérance aux fautes: Possibilité de limiter ou de supprimer les effets d'une perturbation, que celle-ci ait une cause interne ou externe au logiciel. → Traçabilité: Existence de liens entre les différentes représentations textuelles ou graphiques d'un logiciel.
BTD/UML-DL	6

CENTRALE L Y O N	<h2>Les liens entre facteurs et critères</h2>
Bertrand DAVID : UML et le Développement de Logiciels	<ul style="list-style-type: none">● Les exigences, sous forme de facteurs,● La structure du logiciel, par le biais des critères <p>=> il existe une dépendance entre facteurs et critères</p> <ul style="list-style-type: none">→ Confidentialité : Historique des accès, Protection du code et des données en exploitation et hors exploitation→ Couplabilité : Standardisation des données et des interfaces→ Efficacité : Efficacité mémoire, périphériques, temps d'exécution→ Maniabilité : Communicabilité, Exploitabilité, Facilité d'apprentissage→ Robustesse : Précision, Tolérance aux fautes→ Maintenabilité : Lisibilité, Modularité, Observabilité, Simplicité, Traçabilité→ Adaptabilité : Expansibilité, Lisibilité, Modularité, Simplicité, Traçabilité→ Portabilité : Banalité d'emploi, Indépendance par rapport aux environnements logiciel et matériel
BTD/UML-DL	7

CENTRALE L Y O N	<h2>Intérêt de la démarche facteurs-critères</h2>
Bertrand DAVID : UML et le Développement de Logiciels	<ul style="list-style-type: none">● Au niveau de l'expression du besoin (CC): caractéristiques d'utilisation● au niveau de la détermination du processus de développement : Plan d'Assurance Qualité Logiciels (AQL)● Au niveau du contrôle du produit : erreurs de forme et de fond● Problème de facteurs contradictoires :<ul style="list-style-type: none">→ Confidentialité versus Maniabilité→ Efficacité versus Robustesse, Maintenabilité, Adaptabilité, Portabilité
BTD/UML-DL	8