

IFT3903

Qualité du logiciel et métriques

Yann-Gaël Guéhéneuc

Hiver 2006

Chapitre 3 – Qualité du logiciel

(Tiré du cours de Houari Sahraoui)

Plan du cours

1. Introduction
2. Développement logiciel
3. **Qualité du logiciel**
4. Théorie de la mesure
5. Études empiriques
6. Mesure du logiciel
7. Analyse des mesures

Quoi ?

- ISO/IEC 9126 propose 6 caractéristiques de qualité du produit logiciel
 - Capacité fonctionnelle (*functionality*)
 - Fiabilité (*reliability*)
 - Facilité d'utilisation (*usability*)
 - Rendement (*efficiency*)
 - Maintenabilité (*maintainability*)
 - Portabilité (*portability*)
- (D'autres existent !)

Pourquoi ?

■ Situations

- Achat d'un logiciel (alternatives)
- Évaluation de logiciel en cours d'utilisation
- Contrôle de qualité durant le développement d'un logiciel

■ Questions

- le logiciel arrive-t-il à faire ce qui lui est demandé ?
- dans quelle mesure y arrive-t-il ?

Capacité fonctionnelle

■ Définition

- Ensemble d'attributs portant sur l'existence de fonctions et leurs propriétés ; les fonctions sont celles qui satisfont aux besoins exprimés ou implicites

■ Sous-caractéristiques

- Aptitude : présence et adéquation d'une série de fonctions pour les tâches données
- Exactitude : résultats ou effets justes ou convenus
- Interopérabilité : interactions avec d'autres systèmes
- Sécurité : accès non autorisé (accidentel ou délibéré) aux programmes et données

Fiabilité

■ Définition

- ensemble d'attributs portant sur l'aptitude du logiciel à maintenir son niveau de service dans des conditions précises et pendant une période déterminée

■ Sous-caractéristiques

- Maturité : fréquence des défaillances dues à des défauts
- Tolérance aux fautes : aptitude à maintenir un niveau de service donné en cas de défaut ou d'attaque
- Possibilité de récupération : capacité à rétablir son niveau de service et de restaurer les données directement affectées en cas de défaillance ; temps et effort nécessaire pour le faire

Facilité d'utilisation

■ Définition

- Ensemble d'attributs portant sur l'effort nécessaire pour l'utilisation et l'évaluation individuelle de cette utilisation par un ensemble défini ou implicite d'utilisateurs

■ Sous-caractéristiques

- Facilité de compréhension : effort de l'utilisateur pour comprendre la logique et la mise en œuvre
- Facilité d'apprentissage : effort de l'utilisateur pour apprendre son utilisation
- Facilité d'exploitation : effort que doit faire l'utilisateur pour exploiter et contrôler l'exploitation du logiciel

Rendement

■ Définition

- Ensemble d'attributs portant sur le rapport existant entre le niveau de service d'un logiciel et la quantité de ressources utilisées, dans des conditions déterminées

■ Sous-caractéristiques

- Temps : temps de réponses et de traitement ; débits lors de l'exécution de sa fonction
- Ressources : quantité de ressources utilisées ; durée de leur utilisation par fonction

Maintenabilité

■ Définition

- Ensemble d'attributs portant sur l'effort nécessaire pour faire des modifications données

■ Sous-caractéristiques

- Facilité d'analyse : effort nécessaire pour diagnostiquer les déficiences et leurs causes ou pour identifier les parties à modifier
- Facilité de modification : effort nécessaire pour modifier, remédier aux défauts ou adapter à l'environnement
- Stabilité : risque des effets inattendus des modifications
- Facilité de test : effort pour valider le logiciel modifié

Portabilité

■ Définition

- Ensemble d'attributs portant sur l'aptitude du logiciel à être transféré d'un environnement à un autre

■ Sous-caractéristiques

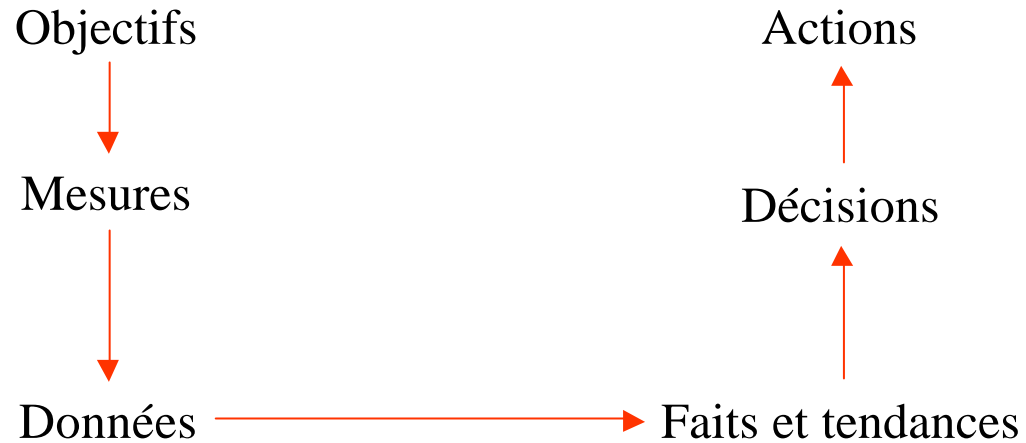
- Facilité d'adaptation : possibilité d'adaptation à différents environnements donnés sans que l'on ait recours à d'autres actions ou moyens que ceux prévus à cet effet par le logiciel
- Facilité d'installation : effort nécessaire pour installer le logiciel dans un environnement donné
- Conformité aux règles de portabilité : conformité aux normes et aux conventions ayant trait à la portabilité
- Interchangeabilité : possibilité et effort d'utilisation du logiciel à la place d'un autre logiciel donné dans le même environnement

Comment ?

■ Concrètement

- La qualité n'est pas une notion unidimensionnelle (voir définitions du chapitre 1)
- Il est donc nécessaire
 - De définir quelles caractéristiques évaluer (quoi)
 - De décider quelles techniques utiliser pour évaluer chacune des caractéristiques (comment)

Comment ?



Aspects mesurables

■ Les processus

- Activités liées au développement du logiciel

■ Les produits

- Objets produits, livrables ou documents qui résultent d'une activité d'un processus

■ Les ressources

- Entités exigées par une activité d'un processus

Aspects mesurables

- Chaque entité des trois classes produits, processus et ressources possède
 - Des attributs internes : attributs mesurables sur l'entité indépendamment de son environnement
 - Des attributs externes : attributs mesurables par rapport aux liens avec son environnement

Aspects mesurables

■ Exemples

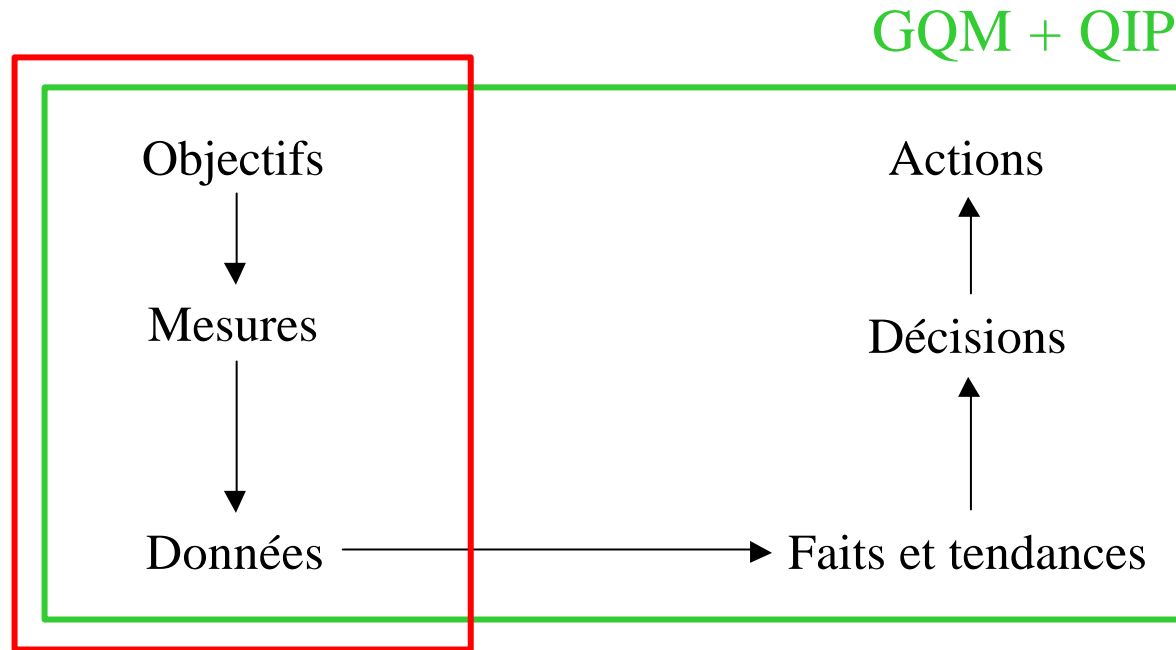
- Attributs internes de processus : effort ou durée du processus ou d'une activité, ...
- Attributs externes de produit : l'efficacité, la portabilité, la facilité de compréhension, ...
- Attributs internes de produit : taille, complexité, couplage, cohésion, ...
- Attributs internes de ressource : personnel, matériels, méthodes, ...

Aspects mesurables

■ Quelques remarques

- Les attributs internes de produits sont souvent utilisés pour prédire les attributs externes
- Ces prédictions permettent de contrôler le développement
- Il est très difficile de définir objectivement des mesures qui dépendent de beaucoup d'autres mesures

Comment ?



ISO9126 + ISO14589

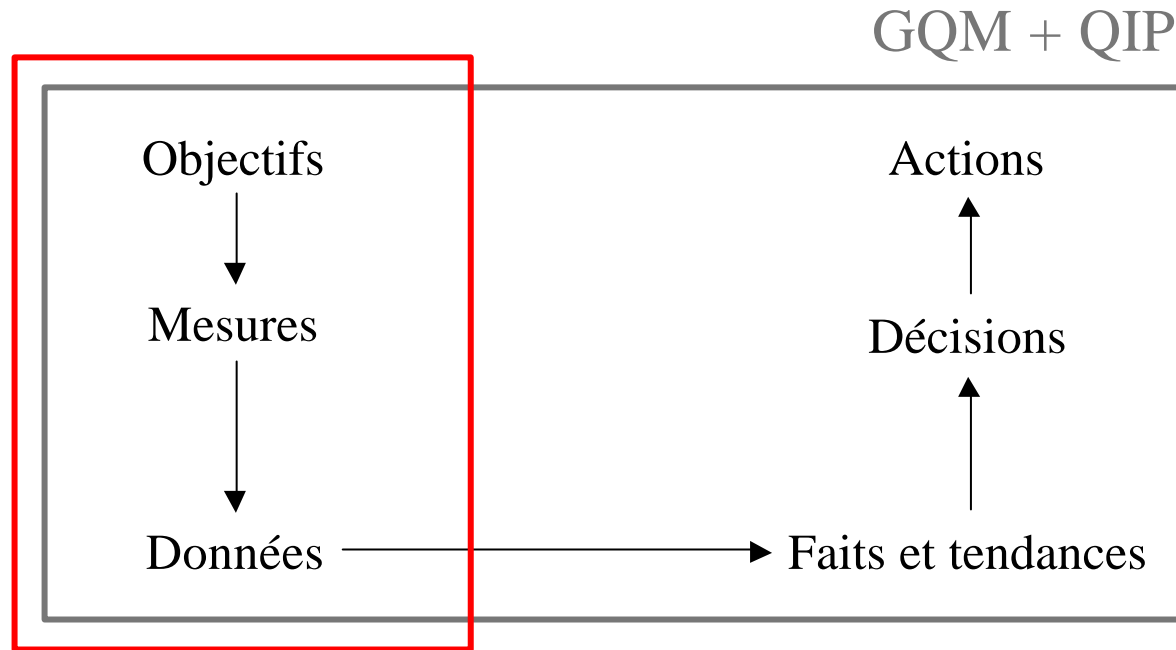
ou

SCOPE

Comment ?

- ISO/IEC 9126 propose également des grandes lignes pour un processus d'évaluation de la qualité
- ISO/IEC 14598 propose un cadre plus précis pour l'évaluation du produit logiciel
- Le projet SCOPE définit un cadre complet pour l'évaluation
- La méthode GQM permet de choisir les indicateurs adéquats et QIP de les améliorer

Comment ?



ISO9126 + ISO14589

ou

SCOPE

Processus d'évaluation (9126)

■ Processus d'évaluation en trois étapes

- La définition des exigences de qualité
 - L'objectif de cette première étape est de spécifier les exigences en termes de caractéristiques de qualité. Ces exigences peuvent varier d'un composant du produit à un autre
- La préparation de l'évaluation
 - À ce niveau, l'objectif est d'initier l'évaluation et de mettre au point ses bases. Ceci est fait en trois sous-étapes

Processus d'évaluation (9126)

- La sélection des indicateurs de qualité. Ces dernières doivent correspondre aux caractéristiques énumérées
- La définition des taux de satisfaction. Les échelles de valeurs doivent être divisées en niveaux de satisfaction des exigences
- La définition des critères d'appréciation. Ceci inclut la préparation de la procédure de compilation des résultats par caractéristique. Il est possible aussi de prendre en compte dans cette procédure des aspects de gestion, tels que le temps ou les coûts
(+ Évaluation multi-critères)

Processus d'évaluation (9126)

– La procédure d'évaluation

- Mesure. Les indicateurs sélectionnés sont appliqués au produit, donnant ainsi des valeurs
- Notation. Pour chaque valeur mesurée, une note (de satisfaction) est attribuée
- Appréciation. En utilisant les critères d'appréciation, un résultat global de l'évaluation du produit est obtenu. Ce résultat est confronté aux aspects de gestion (temps et coûts) pour la prise de décision

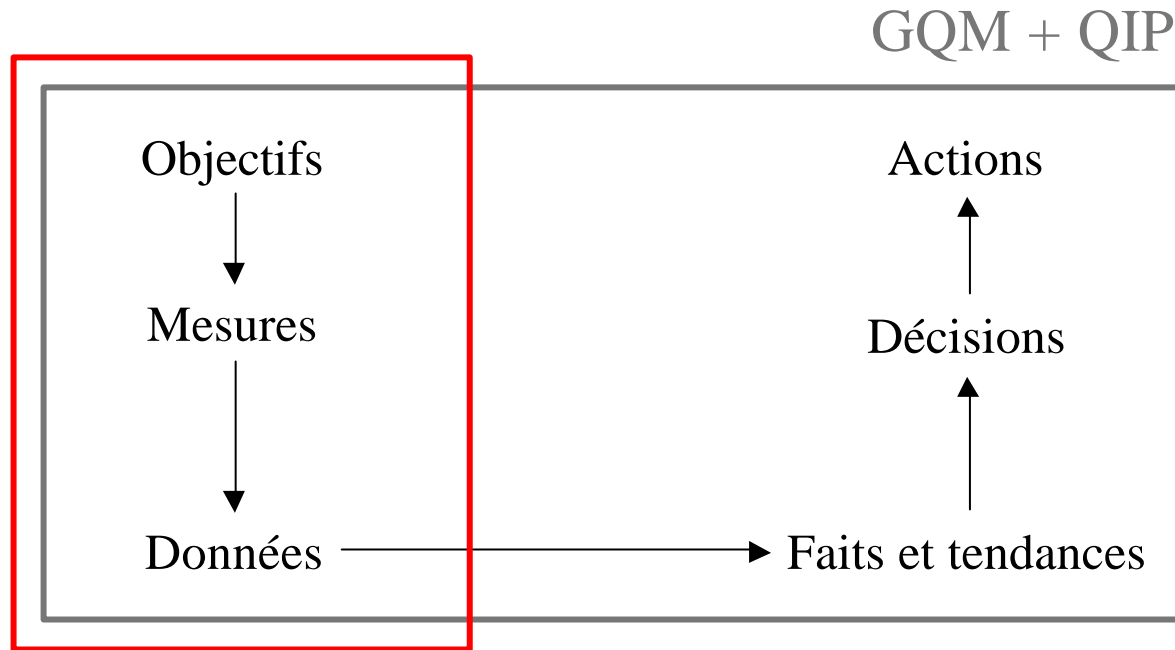
Directives complémentaires (14598)

- L'objectif de cette norme est de fournir
 - Les directives d'identification, d'implantation et d'analyse des indicateurs nécessaires au processus d'évaluation du produit final
 - Les directives de définition des indicateurs qui permettent des évaluations partielles pendant le cycle de développement

Directives complémentaires (14598)

- Cette norme donne entre autres
 - Des informations générales sur des indicateurs de qualité des logiciels
 - Des critères de sélection de ces indicateurs
 - Des directions pour l'évaluation des résultats des mesures (données)
 - Des directions pour l'amélioration du processus de mesure
 - Des exemples de types de graphes d'indicateurs
 - Des exemples d'indicateurs qui peuvent être utilisés pour les caractéristiques de qualité de ISO/IEC 9126

Comment ?



ISO9126 + ISO14589

ou

SCOPE

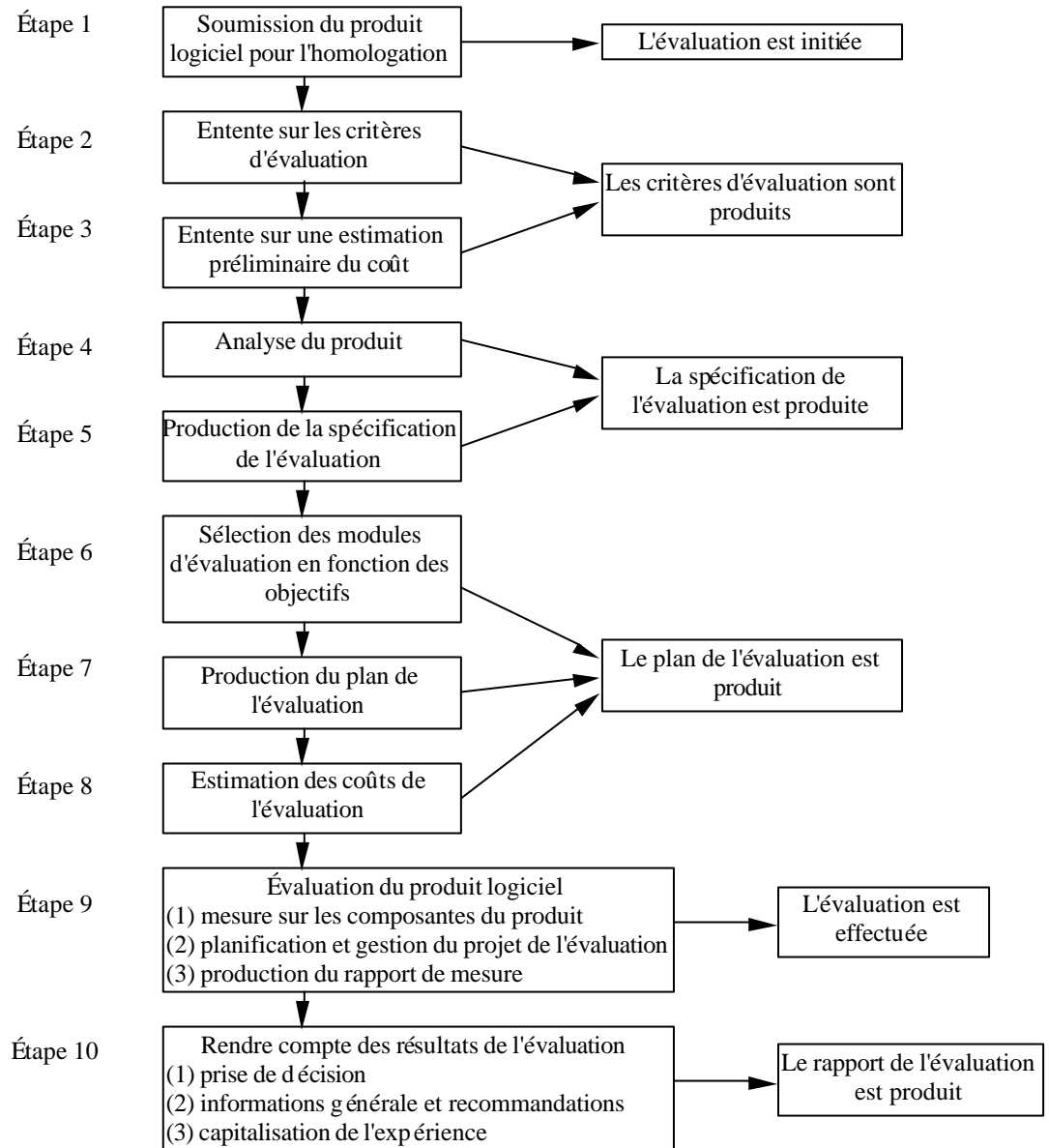
Un cadre d'évaluation, SCOPE

- SCOPE est un projet européen ESPRIT (1989-93)
(*Software CertificatiOn Programme in Europe*)
- Objectifs
 - Définir des procédures d'attribution d'un label de qualité à un logiciel quand celui-ci satisfait un certain ensemble d'attributs de qualité
 - Développer des technologies nouvelles et efficaces d'évaluation, à des coûts raisonnables, permettant l'attribution de ce label
 - Promouvoir l'utilisation des technologies modernes de l'ingénierie des logiciels. Celles-ci, utilisées durant le développement, contribuent à l'attribution du label

Un cadre d'évaluation, SCOPE

- Résultat : définition d'un cadre d'évaluation comprenant
 - Un processus
 - Une méthode
 - Des techniques

Processus SCOPE



Processus SCOPE

- Documents produits
 - Les critères d'évaluation
 - La spécification de l'évaluation
 - Le plan de l'évaluation
 - Le rapport d'évaluation

Méthode SCOPE

- La méthode d'évaluation s'appuie sur trois types d'analyse techniques
 - L'analyse statique qui consiste à examiner le code pour évaluer les caractéristiques de qualité
 - L'analyse dynamique qui consiste entre autres à simuler le déroulement du programme pour effectuer des mesures
 - L'inspection qui concerne particulièrement les interfaces personnes–machines

Méthode SCOPE

- L'évaluation peut se faire selon le niveau de détail souhaité

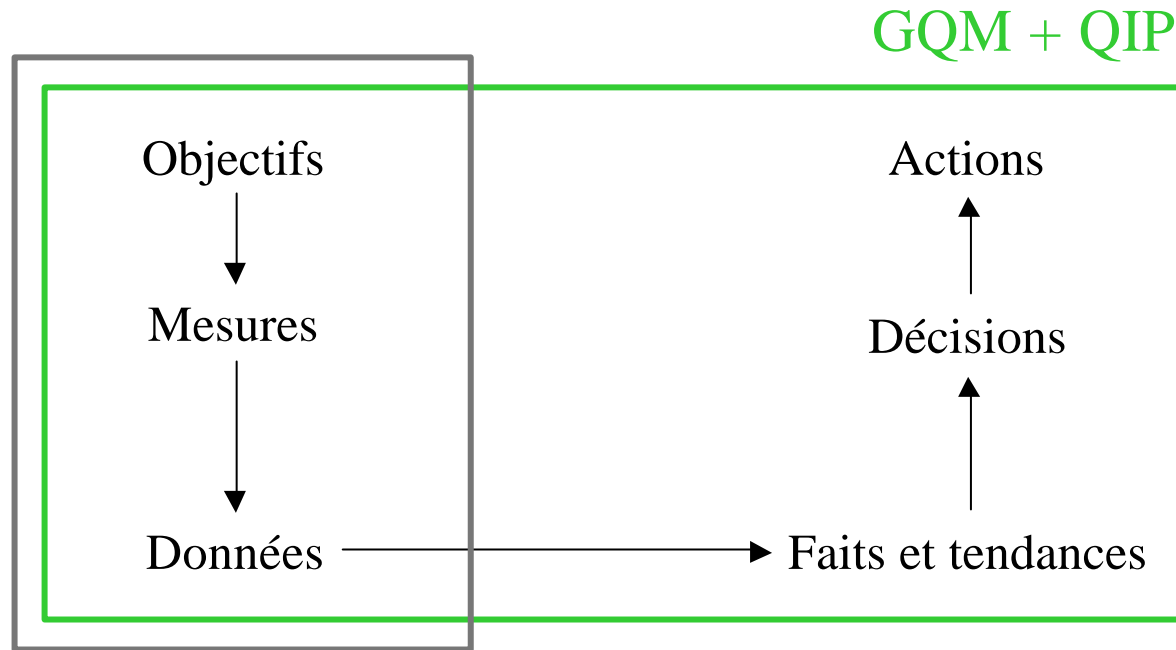
Niv.	Environnement	Personnes	Économie	Application
D	petit dommage à la propriété	pas de risques pour les personnes	perte économique négligeable	loisirs, domestiques
C	dommage à la propriété	peu de personnes touchées	perte économique significative	alarmes de feu, contrôle de processus
B	dommage environnemental réparable	menace pour des vies humaines	grande perte économique	systèmes médicaux, systèmes financiers
A	dommage environnemental irréparable	décès de personnes	désastre financier	systèmes de transport, systèmes du nucléaire

Techniques SCOPE

■ Choix des techniques pour chaque niveau

	Niveau D	Niveau C	Niveau B	Niveau A
Capacité fonctionnelle	test fonctionnel (boîte noire)	+ inspection des documents (listes de contrôle)	+ test des composantes	+ preuve formelle
Fiabilité	facilités des langages de programmation	+ analyse de la tolérance aux fautes	+ modèle de croissance de la fiabilité	+ preuve formelle
Facilité d'utilisation	inspection des interfaces utilisateur	+ conformité aux normes sur les interfaces	+ test en laboratoire	+ modèle mental de l'utilisateur
Rendement	mesurage du temps d'exécution	+ test avec bancs d'essais (<i>benchmarks</i>)	+ complexité algorithmique	+ analyse des performances
Maintenabilité	inspection des documents (listes de contrôle)	+ analyse statique	+ analyse du processus de développement	+ évaluation de la traçabilité
Portabilité	analyse de l'installation	+ conformité avec les règles de programmation	+ évaluation des contraintes de l'environnement	+ évaluation de la conception des programmes

Comment ?



ISO9126 + ISO14589

ou

SCOPE

Problème : le choix d'une mesure

- On ne mesure pas pour le plaisir de mesurer
- Comment choisir la bonne mesure quand vient le temps de mesurer ?
- Le choix de la mesure dépend de l'objectif des mesures
- L'une des approches les plus utilisées pour le choix des mesures est GQM (*Goal – Question – Metrics*)

GQM : introduction

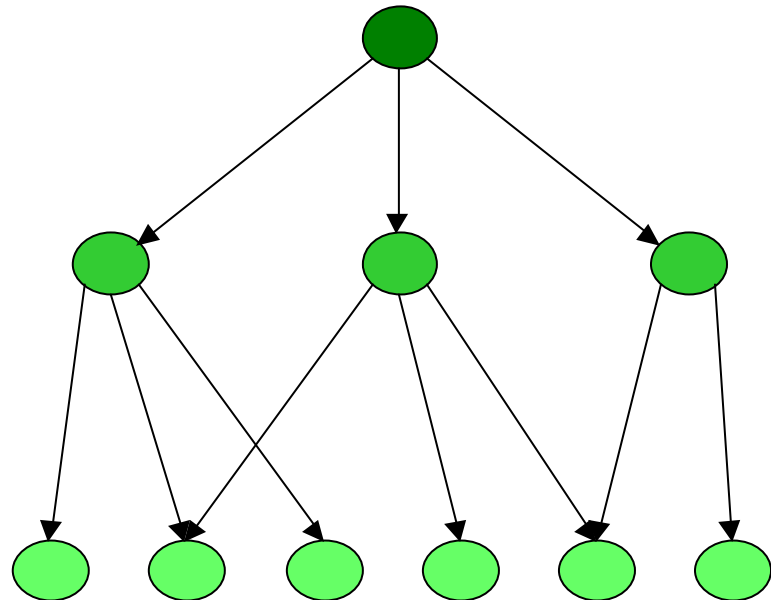
- Approche introduite par Basili *et al.*
- GQM propose un cadre en trois étapes
 - Énumérez les objectifs principaux du projet de développement ou de maintenance
 - Dérivez de chaque objectif, les questions dont les réponses permettent de déterminer si le but est atteint
 - Décidez de ce qui doit être mesuré afin de pouvoir répondre aux questions

Aperçu

Objectif (*goal*)

Questions

Indicateurs (*metrics*)



Exemple

Objectif

Identifier très tôt les
modules générateurs
d'erreurs

Questions

Que veut dire modules
générateurs d'erreurs ?

Est ce que la complexité a une
influence sur la génération
d'erreurs ?

Combien de tests seront
effectués par module ?

Métriques

Données sur les fautes de chaque module
Nombre de fautes par phase de test
Nombre d'échecs attribués à un module

*Données sur la taille et la
complexité de chaque module*
LOC
Mesure de complexité

*Données sur l'effort pour
chaque module*
L'effort par phase de test
**Nombre d'échecs attribués à
un module**

GQM : composantes de l'approche

■ Paradigme

- Définit les principes à suivre

■ Plan (modèle)

- Décrit l'objectif des mesures, les questions dérivées et les mesures qui en découlent
- Définit quelles mesures utiliser et pourquoi

■ Méthode

- Donne les lignes directrices pour initier et exécuter des programmes de mesure

Paradigme GQM

- Le paradigme de GQM est basé sur l'idée que la mesure doit être guidée par un objectif
- Toute collecte de données dans un programme de mesure doit être basée sur un raisonnement explicitement documenté
- Avantages
 - Aide dans l'identification des indicateurs utiles et appropriés et dans l'analyse et l'interprétation des données collectées
 - Permet une évaluation de la validité des conclusions tirées et évite les rejets des programmes de mesure

Paradigme GQM

■ Principes

- La tâche d'analyse à exécuter doit être spécifiée avec précision et de manière explicite (objectif explicite de la mesure)
- Chaque indicateur doit avoir une justification explicitement documentée ; cette justification est utilisée pour expliquer la collecte des données et pour guider l'analyse et l'interprétation de ces données
- Les personnes qui définissent l'objectif de la mesure doivent être complètement impliquées dans l'initiation et l'exécution du programme de mesure

Plan GQM

- Le plan décrit en détail l'analyse basée la mesure
- Il comporte trois niveaux de raffinement
 - Niveau conceptuel : un objectif est défini pour une entité, en fonction d'un modèle de qualité, par rapport à une point de vue dans un environnement donné
 - Niveau opérationnel : un ensemble de questions est utilisé pour définir quantitativement l'objectif et spécifier comment cet objectif sera interprété
 - Niveau quantitatif : un ensemble de données est associé à chaque question pour permettre d'y répondre de manière quantitative

Plan GQM

- Un objectif doit préciser
 - Quelle entité est analysée
 - L'objectif de l'analyse
 - Quelle caractéristique est analysée
 - Le point de vue qui doit guider l'analyse
 - L'environnement de l'analyse
- Catégorisation des questions
 - Modèles de qualité
 - Facteurs d'influence

Méthode ou processus GQM

- Il n'existe pas une façon standard d'appliquer l'approche GQM
- Un exemple de processus est celui en 7 étapes
 - Caractérisation de l'organisation et du projet
 - Identification des objectifs de la mesure
 - Production du plan GQM
 - Production du plan de mesure
 - Collecte et validation des données
 - Analyse des données et interprétation
 - Stockage des résultats à fins de réutilisation

La mesure pour l'amélioration

- Mesure = Amélioration systématique
- Relation entre mesure et amélioration
 - La mesure décrit quantitativement l'état courant
 - La connaissance de l'état courant permet de définir des objectifs quantitatifs réalistes d'amélioration
 - La connaissance de l'état actuel permet d'identifier les points forts et les points faibles du processus utilisé
 - La connaissance des points faibles du processus permet d'identifier les changements à faire pour l'améliorer
 - L'impact d'un changement ne peut être mesuré que s'il existe une base quantitative permettant la comparaison

La mesure pour l'amélioration

- L'approche GQM fait partie d'une approche globale appelée QIP (*Quality Improvement Paradigm*)
- Processus d'amélioration en 6 étapes
 - Caractérisation
 - Définition des objectifs
 - Choix du processus
 - Exécution
 - Analyse
 - Consolidation

Exemple de plans de qualité GQM

- AT&T

- Motorola

- Voir <http://ieeexplore.ieee.org/search/wrapper.jsp?arnumber=268959>

Exemple de plans de qualité GQM

■ AT&T

- AT&T used GQM to help determine which metrics were appropriate for assessing their inspection process [Barnard and Price 1994]

Exemple de plans de qualité GQM

■ AT&T

Goal	Question	Metrics
Plan	How much does the inspection process cost?	Average effort per KLOC Percentage of reinspections
	How much calendar time does the inspection process take?	Average effort per KLOC Total KLOC inspected
Monitor and control	What is the quality of the inspected software?	Average faults detected per KLOC Average inspection rate Average preparation rate
	To what degree did the staff conform to the procedures?	Average inspection rate Average preparation rate Average lines of code inspected Percentage of reinspections
	What is the status of the inspection process?	Total KLOC inspected
Improve	How effective is the inspection process?	Defect removal efficiency Average faults detected per KLOC Average inspection rate Average preparation rate Average lines of code inspected
	What is the productivity of the inspection process?	Average effort per fault detected Average inspection rate Average preparation rate Average lines of code inspected