

Réseau Mixte Technologique
Modélisation et logiciels d'intérêt commun appliqués à l'agriculture

Validation des modèles et outils pour l'agronomie

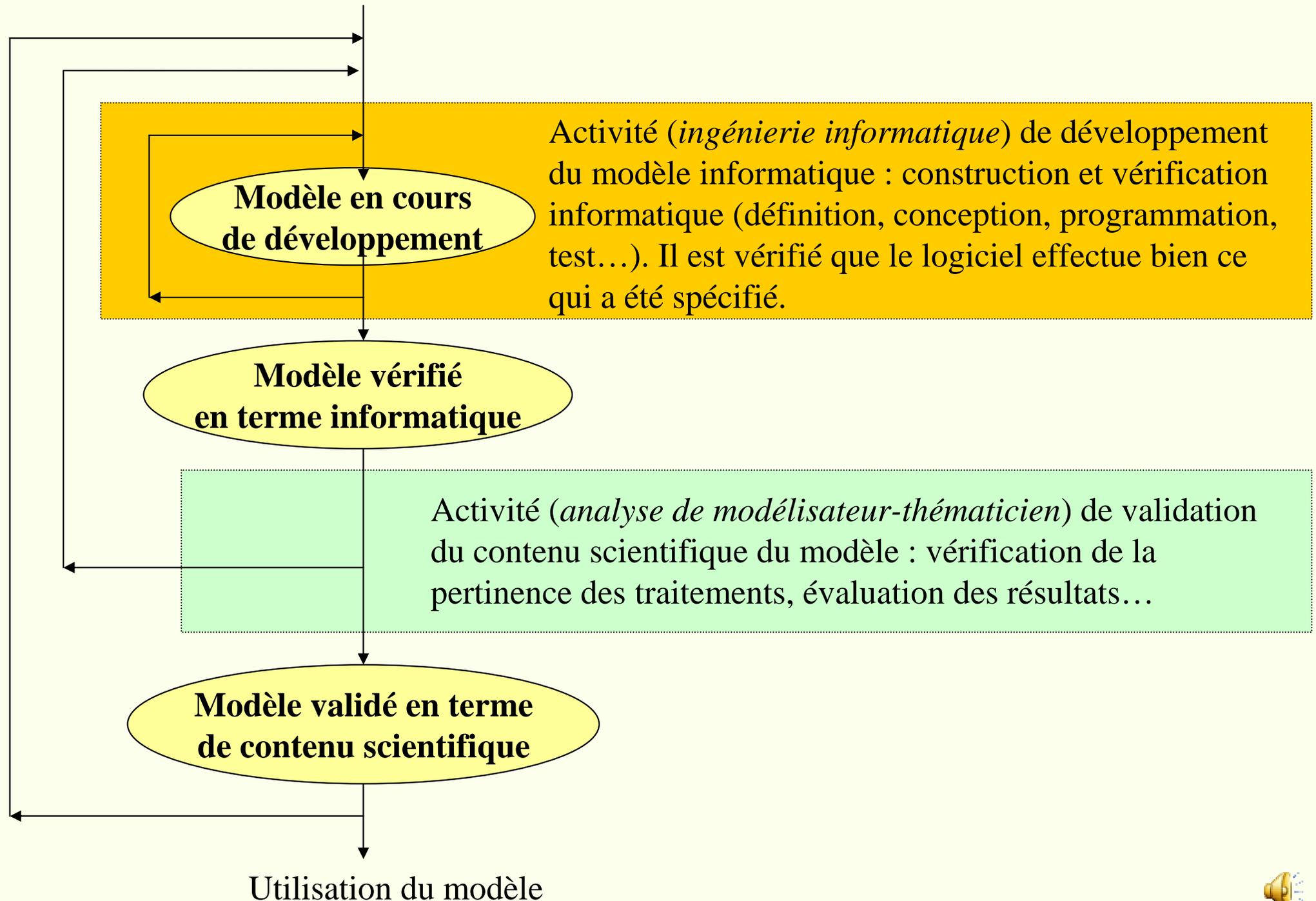
16 novembre 2009

Vérification informatique du modèle, tests logiciels

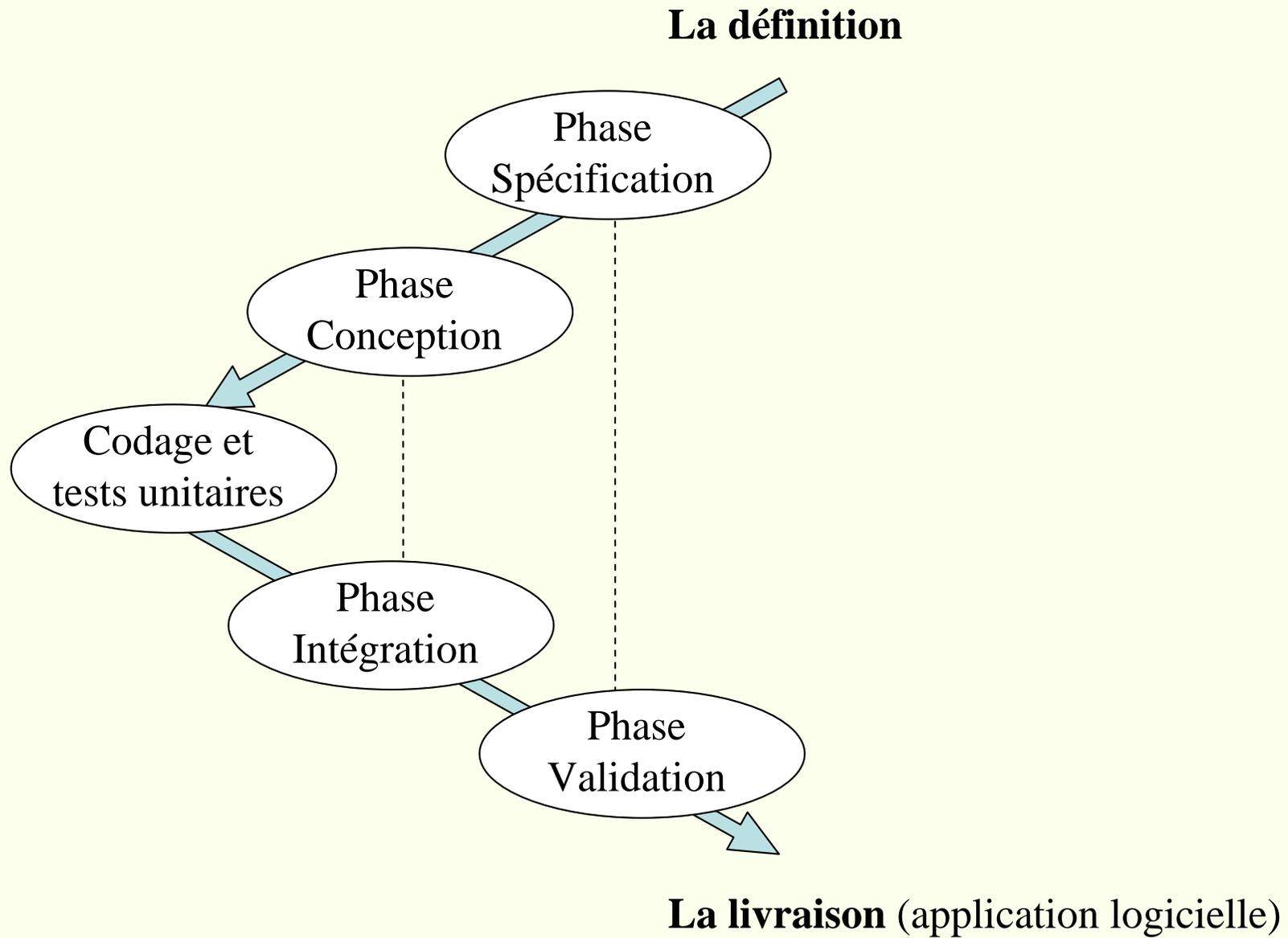
Nathalie Rouse, INRA, <http://www.modelia.org>



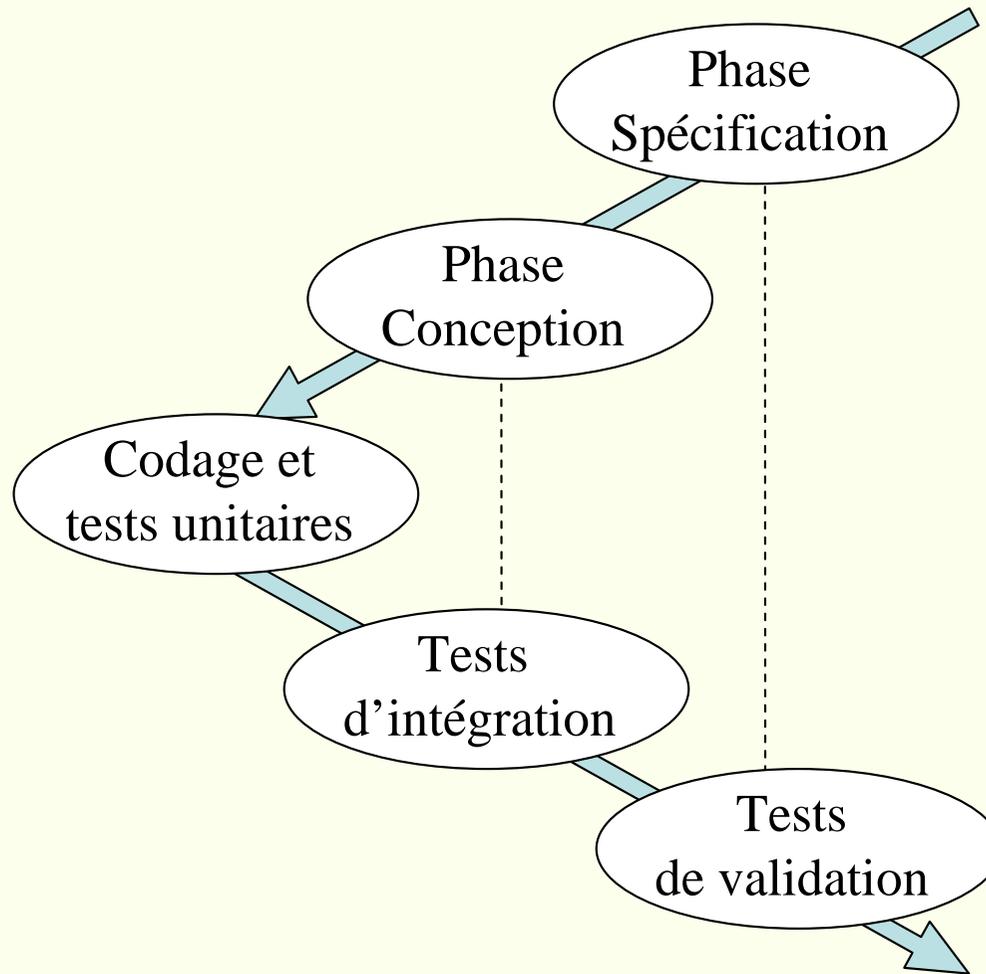
Stades de développement d'un modèle informatique



Cycle en V



Les tests logiciels



... Tests en environnement réel

Spécification

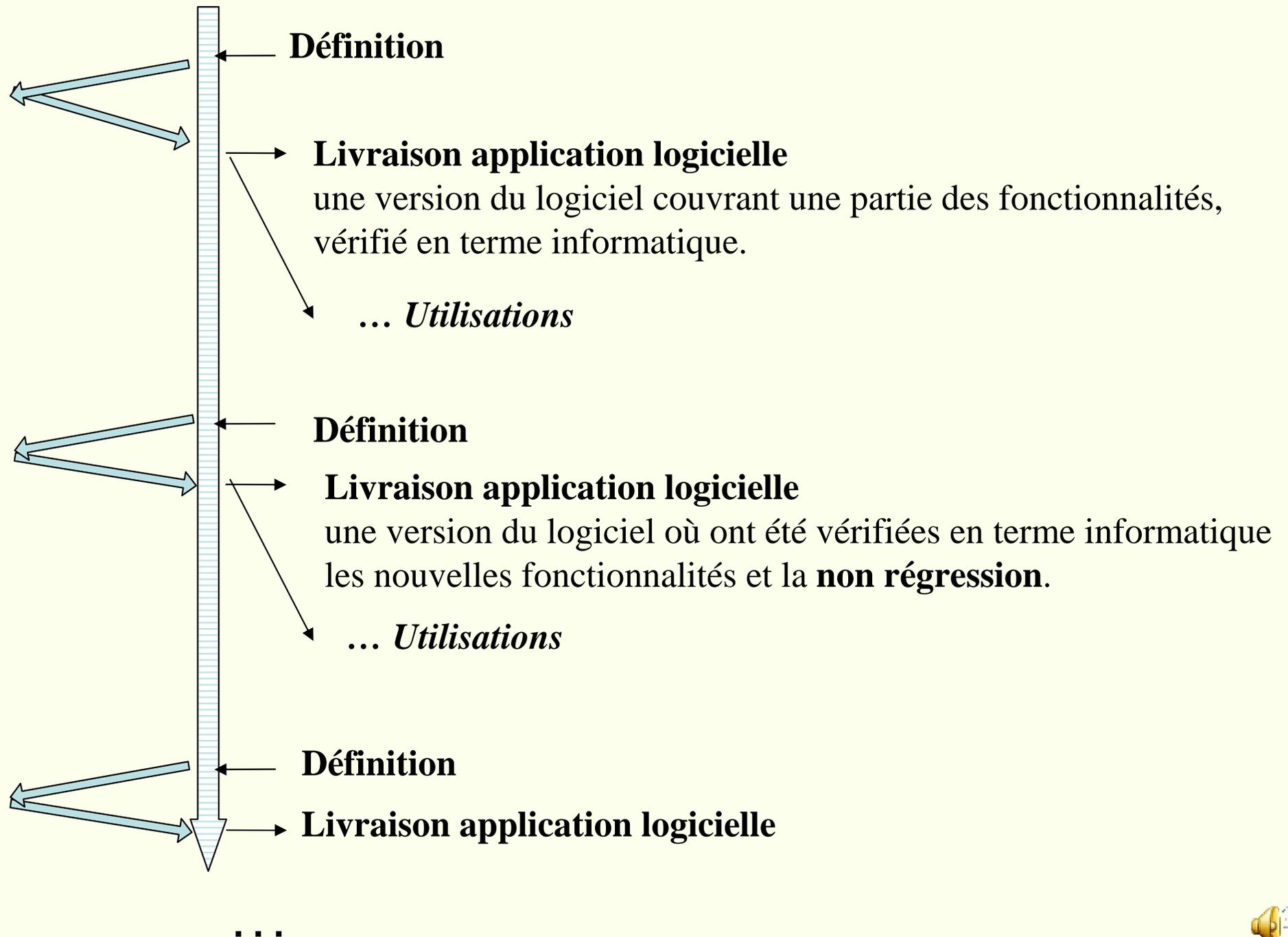
Définition de ce que doit faire le logiciel

Validation logicielle

Garantie de ce que fait le logiciel, fiabilité, stabilité, non régression



Développement incrémental



Formalisation des tests

Les documents de test définissent les tests et leur mise en œuvre, rendent compte des résultats des tests. La formalisation facilite la vérification de la **non régression**.

Le **plan de test** du logiciel définit l'organisation et l'environnement de mise en œuvre des tests, décrit la prévision et la planification des tests.

Le **document de description des tests** du logiciel sert de support à l'exécution des tests. Il décrit les procédures et les outils à mettre en œuvre. Il définit les **jeux d'essais**.

Le **rapport des tests** du logiciel enregistre, pour chaque jeu d'essai, les résultats des tests : test OK ou not OK, observations en cours de tests.



Que tester ?

Tests fonctionnels : tests de cas nominaux, cas aux limites...

Tests boîte noire, boîte blanche

Tests de configuration/d'installation

Tests de compatibilité/d'interopérabilité

Tests d'interface

Tests de performance, stress, volume

Tests de robustesse

Tests de convivialité

...



Comment tester ?

Ecriture de programme de test, scripts de tests...

La librairie Boost <http://www.boost.org>, librairie C++ libre, fournit des fonctions permettant de programmer des tests : BOOST_(REQUIRE/CHECK)_(CLOSE/SMALL/EQUAL).

Débogueur.

Revue logicielles, vérification/relecture de code par un tiers.

Automatisation des tests



Un jeu de test

→ Injection des conditions initiales, des entrées

→ Exécution du logiciel à tester

→ Capture des résultats, des sorties

→ Vérifications :

`BOOST_REQUIRE_CLOSE(valeur resultat, valeur attendue, pourCent);`

→ Notification des résultats, conclusions du test dans le rapport de test.



Un exemple de mise en œuvre de tests logiciels sur <http://www.modelia.org> :

Tests du logiciel portageMM2CPP de portage de code ModelMaker en langage C++

(sous la rubrique « Ingénierie d'un projet informatique / Logiciels diffusés »).

