Spécification et validation du logiciel

Jean Goubault-Larrecq

G.I.E. Dyade, INRIA

Jean.Goubault@dyade.fr

http://www.dyade.fr/fr/actions/vip/jgl.html

Buts de ce cours

- Savoir décrire ce qui est attendu d'un programme : *spécification*.
- Savoir décrire ce que fait réellement un programme : *modélisation*.
- Savoir vérifier que le programme fait réellement ce qui est attendu : *vérification*, *preuve*.

Pour quoi faire ?

- pour éviter les bugs de conception;
- pour faciliter la maintenance du programme;
- pour servir de base à la documentation du programme;
- pour certifier le programme auprès d'autorités;
- pour servir à la conception de jeux de tests; etc.

Exemples de domaines d'application

- Sécurité ferroviaire/aérienne (absence de collisions);
- Protocoles cryptographiques (sécurité, confidentialité, authentification), commerce électronique;
- Conception de circuits (cf. model-checking);
- OS multi-processeurs (cohérence des caches);
- Bases de données (cohérence des informations);
- Conception de langages de programmation (absence d'erreurs de type à l'exécution);

Quelques exemples de bugs

- Ariane 5 (débordement numérique dans un code inutile; problèmes de deadlocks);
- Thérac 25 (mauvaise gestion d'unités de mesure);
- Bug F00F du Pentium (interférence bus lock/illegal instruction);
- Protocoles de Needham-Schroeder, d'Otway-Rees buggés (cas de figures négligés);