

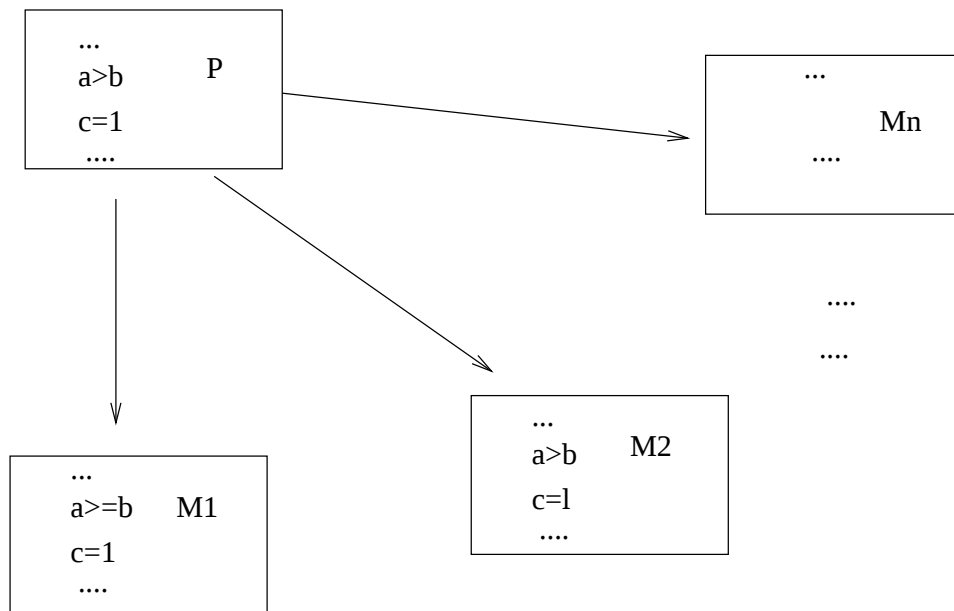
3. Técnica Baseada em Erros

- Utiliza informações sobre o processo de desenvolvimento de software.
- Utiliza os tipos mais comuns de erros cometidos por programadores
- Análise de Mutantes

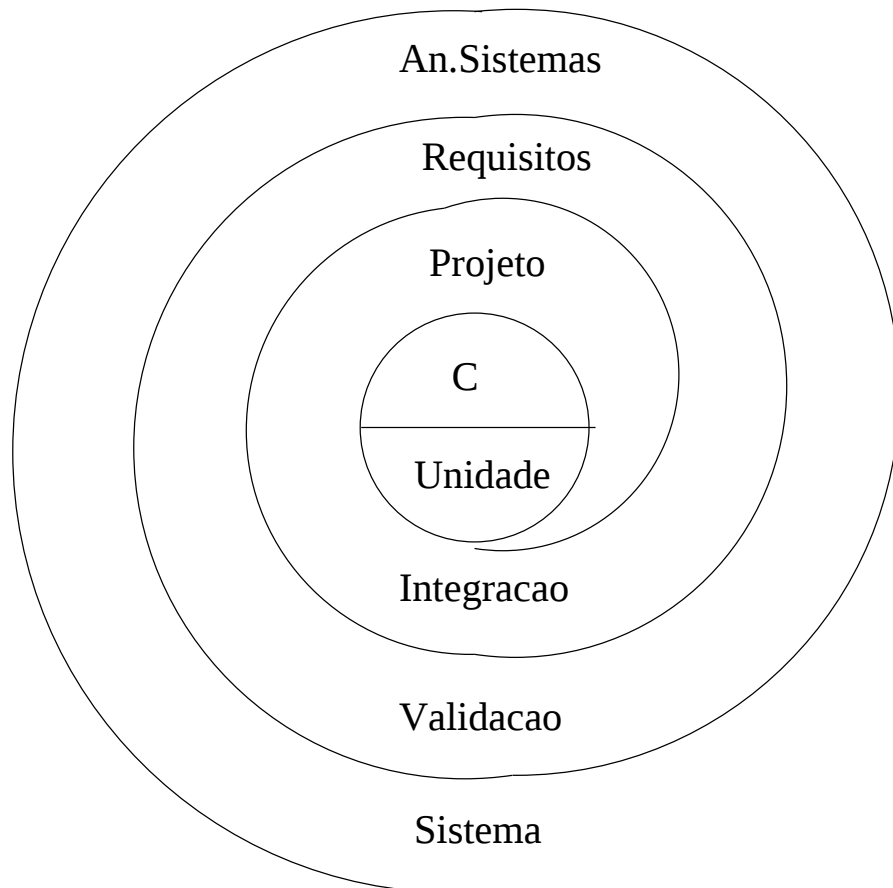
3.1 Análise de Mutantes

Mutantes diferem ligeiramente do programa original

Objetivo: fazer com que eles se comportem diferentemente.



VII- Uma Estratégia para Teste



1. Teste de Unidade

- Tem como objetivo testar a menor unidade do projeto: o módulo.
- Visa a examinar a estrutura de dados local, identificar erros de lógica e de implementação.
- As técnicas estruturais são as mais utilizadas.

2 Teste de Integração

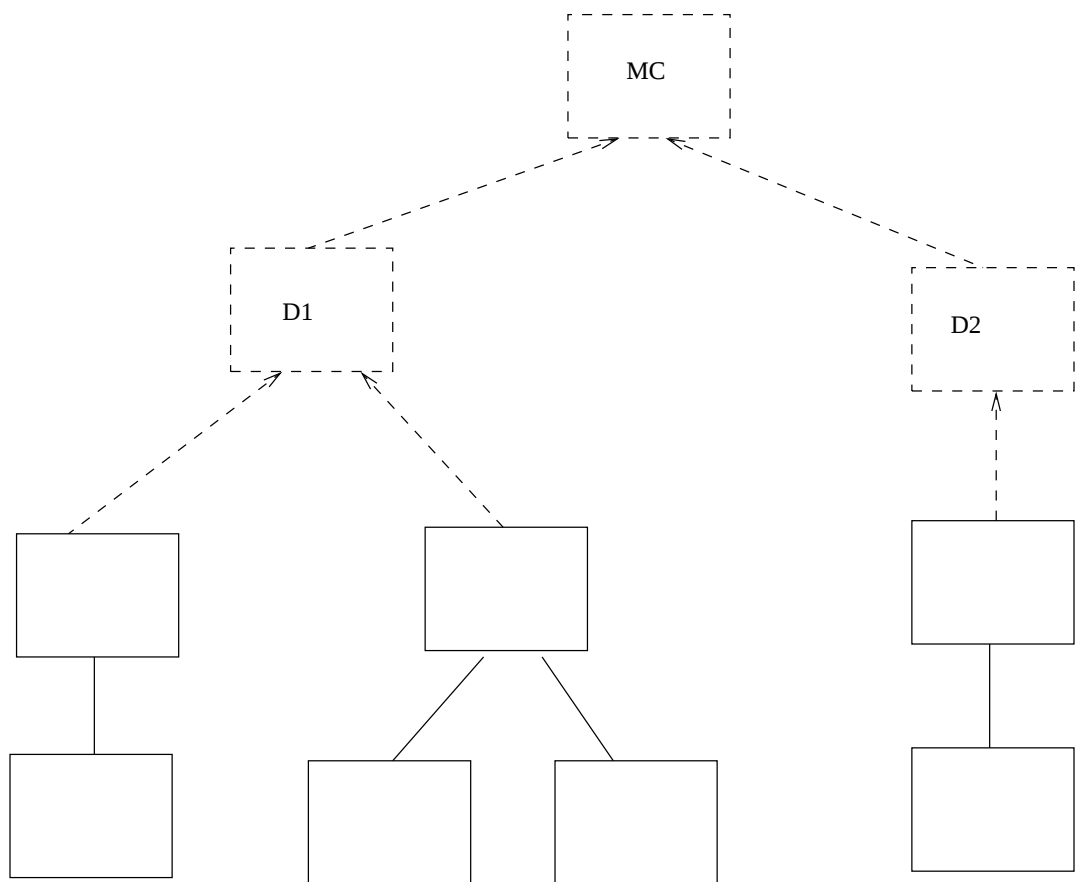
- Visa a integrar os módulos componentes na estrutura do software.
- Visa a identificar erros de interface entre os módulos.
- Teste não-incremental: testa tudo de uma vez.
- Teste incremental: testa segmentos da estrutura: top-down e bottom-up.

3 Teste de Validação

- Permite testar o programa de acordo com os requisitos funcionais e de desempenho.
- Técnicas funcionais são as mais utilizadas.
- *Teste alpha*: testes realizados com o usuário num ambiente controlado pelo desenvolvedor.
- *Testa beta*: testes realizados pelo usuário em seu ambiente; não controlado pelo desenvolvedor.

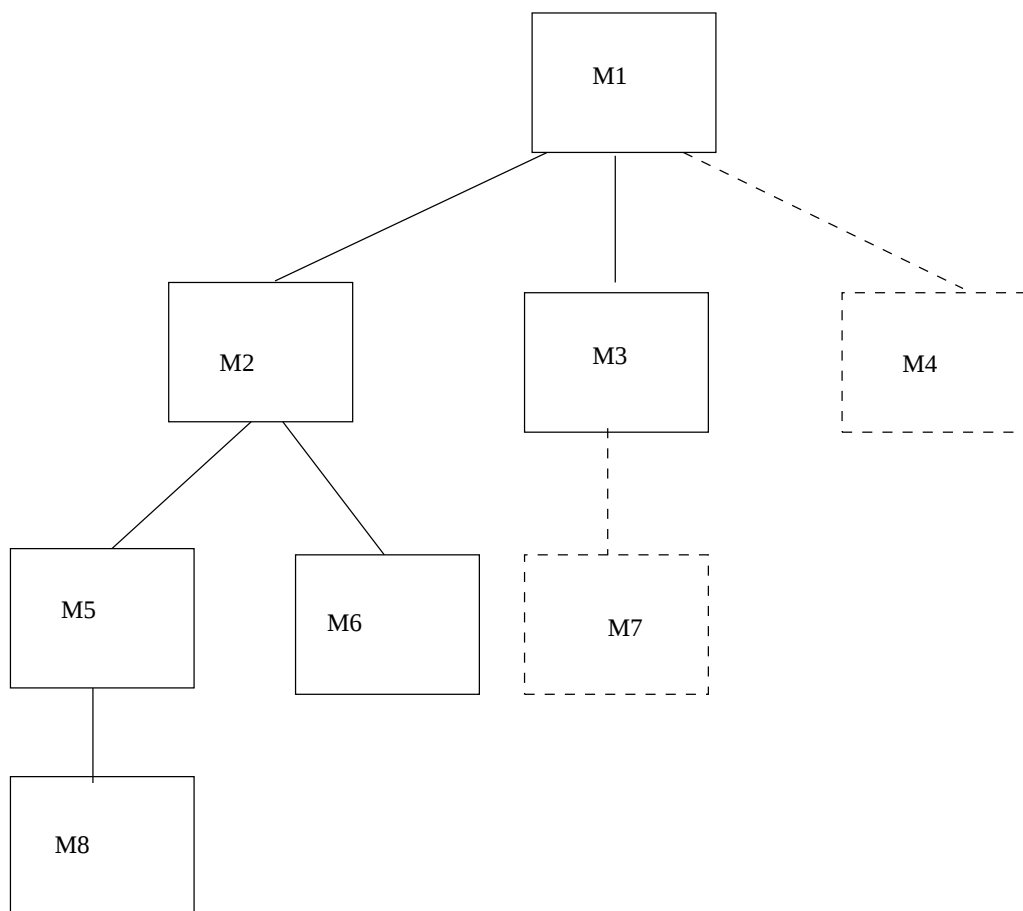
Integração Bottom-Up

Os drivers vão sendo substituídos e sendo combinados



Integração Top-Down

Os stubs vão sendo substituídos por programas reais



4 Teste de Sistema

- Visa a testar o software com todos os outros elementos do sistema.
- Testa função e desempenho.
- Testa tolerância a falhas.
- Testa segurança.
- Testa o sistema em situações anormais.