

PLoPime 2006 Anti-Padiao de Desenvolvimento: Desastre Incomensuravel

Autores: Mario Marques Junior e Reginaldo Rideaki Kamiya

Jean Claude: – Ei, John. Estou fazendo o monitoramento da sonda espacial e temos um problema. Nem mesmo os dados da velocidade estao batendo, e a sonda esta completamente fora da rota planejada.

John Smith: – Absolutamente nao, caro Jean Claude. Tambem estou monitorando daqui, e todos os dados de velocidade, desde o lancamento, estao de acordo com o planejado. Por conta disso, a rota que tenho aqui no meu esquema tambem esta coincidindo.

Jean Claude: -- Desculpe-me insistir, mas voce pode me dizer qual a velocidade atual da sonda segundo o seus dados de monitoramento?

John Smith: – Claro! 138 pes/s, com desvio de 0,5 pes/s e probabilidade 0,99.

Jean Claude: -- M! 138 PES POR SEGUNDO? Bom, com esse dado que voce acaba de me passar, so posso afirmar que... ..hum... ..em 1 minuto e 13 segundos aproximadamente... ..perderemos o contato com a sonda, pois ela esta somente a alguns poucos METROS de distancia do limite toleravel...*

Raizes do Problema

Um defeito no codigo, em sistemas de alta confiabilidade e custo, e encontrado apenas quando ele ja esta em uso.

Exemplo

Inumeros casos envolvendo a NASA, e o Departamento de Seguranca dos Estados Unidos, entre outros.

Sintomas e Consequencias

Uma consequencia imediata e a falha do projeto como um todo, e desperdicio de recursos. As consequencias seguintes dessa falha e gerar um descredito em sistemas desse tipo, fazendo com que seja muito mais dificil aprovar um novo projeto desse porte; a nao ser que se invista em testes, o que torna o projeto muito mais caro. A despeito desse investimento, o Desastre Incomensuravel continua a ocorrer, fazendo com que o orcamento para testes suba ainda mais, inviabilizando uma nova tentativa. E uma bola de neve.

Causas Típicas

Não existe uma causa definitiva para que o Desastre Incomensurável ocorra, mas como prevenir e melhor do que remediar, nesse caso, é bom saber o que pode estar causando esse Anti-Padrão.

Primeiro, porque os desenvolvedores continuam usando medidas que não são compatíveis? Possivelmente:

- Esse tipo de projeto envolve desenvolvedores de vários países. Um desenvolvedor de um determinado país pode se sentir mais vontade trabalhando com unidades e constantes familiares, principalmente em magnitudes que ele consegue ver/sentir.
- As medidas usadas não são consenso entre os desenvolvedores, principalmente quando foi decidida por superiores e há falha de comunicação.

Depois, porque os defeitos continuam no sistema, mesmo se investindo em testes? Pode ser que:

- Os desenvolvedores testam seus componentes isoladamente, e não há um teste que envolva todos os componentes do sistema, ou;
- Mesmo havendo um teste de unidade, ele também é incoerente em relação às medidas.

Solução Refatorada

- Deixe os desenvolvedores acordarem quais as unidades e constantes usadas no sistema. Para equipes em um mesmo local, use quadro negro para fixar o que foi acordado e facilitar a comunicação. Para equipes distribuídas, use um quadro virtual, mas tenha certeza que todos o consultam frequentemente.
- Tenha uma equipe centralizada para desenvolver testes de unidade. Um contra dessa medida é impedir que o desenvolvedor local faça seus próprios testes. Para resolver isso podem ser criadas duas camadas de testes: uma centralizada e que abrange todo o sistema, e abaixo dessa uma distribuída para componentes (ou grupos de componentes), em que os testes podem ser temporariamente incoerentes em relação ao sistema como um todo.

Exceções Conhecidas

Não se aplica a esse Anti-Padrão.

Soluces Relacionadas

- A falta de testes que provavelmente pode estar causando esse Anti-Padrão tem relação com Blind Faith (fe cega).
- A construção e uso de glossários no sistema é uma solução semelhante.