Trabalho Supervisionado de Formatura

Dívida Técnica

Identificando, Medindo e Monitorando

Diogo de Jesus Pina
Orientador: prof. Dr. Alfredo Goldman

Agenda

- Dívida Técnica
  - O que é?
  - Classificações
- Identificando Dívida Técnica
  - Indicadores Diários
  - Ferramentas
- Medindo a Dívida Técnica
  - Fórmula
  - Modelos de Qualidade
  - Sonar Qube
  - Armazenamento
- Monitorando a Dívida Técnica
  - Tipos de Monitoramento
  - Arcabouço de Gerenciamento
  - Técnicas para Evitar a Dívida Técnica

O que é dívida técnica?

"Dívida técnica é uma metáfora para artefatos imaturos, incompletos ou inadequados no ciclo de vida de desenvolvimento de software."

Carolyn Seaman e Yuepo Guo em
Mensuring and Monitoring Technical Debt

Classificação Dívida Técnica

- Intencional e
- Não Intencional.
  - Curto Prazo e
  - Longo Prazo.

Identificando a Dívida Técnica

Indicadores Diários de Dívida Técnica

- "Não se preocupe com a documentação para agora"
- "O único que pode alterar este código é o Felipe."
- "Isto está bom agora, mas depois precisa refatorar."
- "ToDo/FixMe"
- "Vamos apenas copiar e colar esta parte."
- "Alguém sabe onde guardamos a senha do banco de dados?
- "Eu sei se eu tocar neste código alguma coisa vai quebrar."
- "Vamos terminar os testes no próximo *release*."
- "O *layout* só esta funcionando para celulares com baixa resolução."

Padrões de Código

- Números Mágicos;
- Padrões de sintaxe;
- Erros potenciais;
- Comentários de linhas de código e
- Atributos não utilizados.

http://findbugs.sourceforge.net

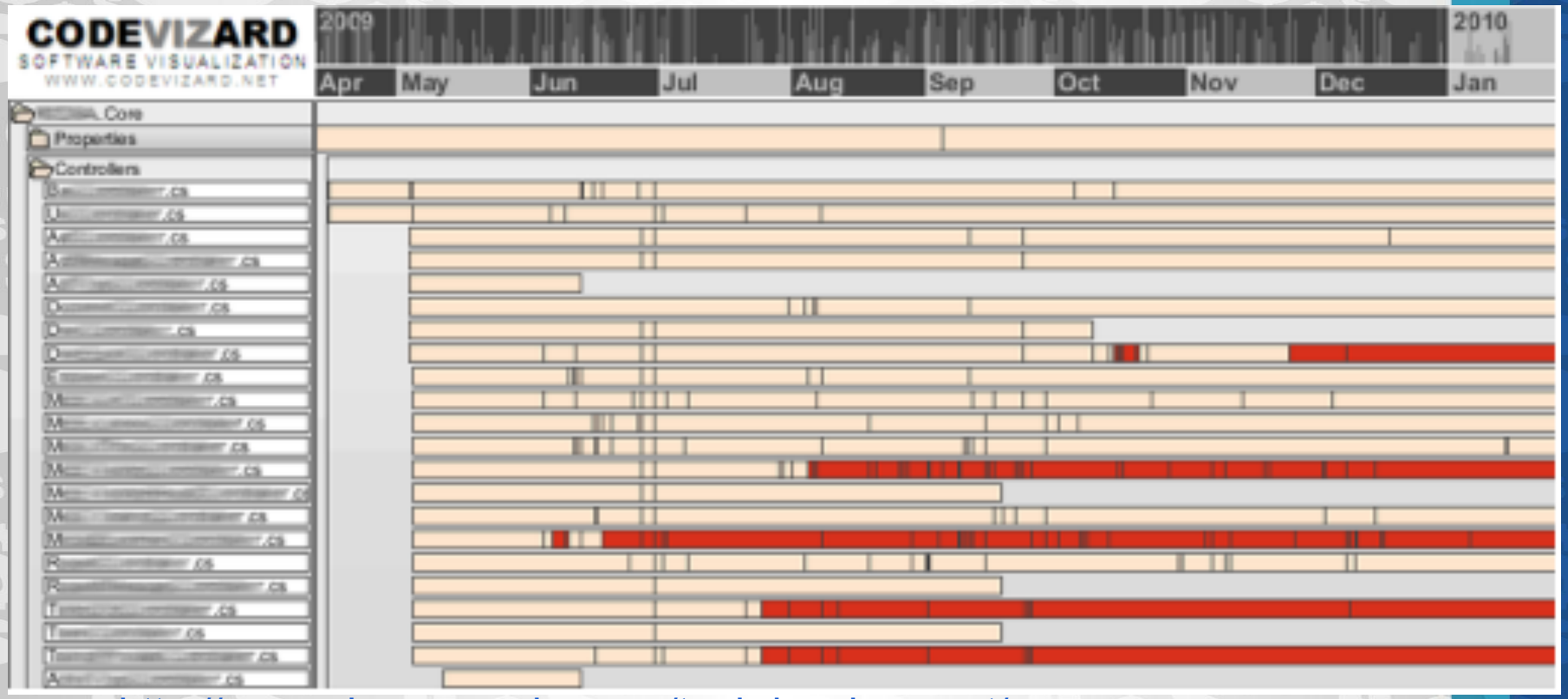
Code Smells

Métodos e classes que violam os princípios do bom design de orientação a objetos.

- Classes ou Métodos Longos;
- Métodos com muitos parâmetros;
- Código duplicado;
- Complexidade Condicional;
- Explosão combinatorial;
- Nomes de variáveis, métodos e classes que não possuem significado;
- Código Morto e
- Generalidade Especulativa.

"If it stinks,change it."
Kent Beck

Code Vizard

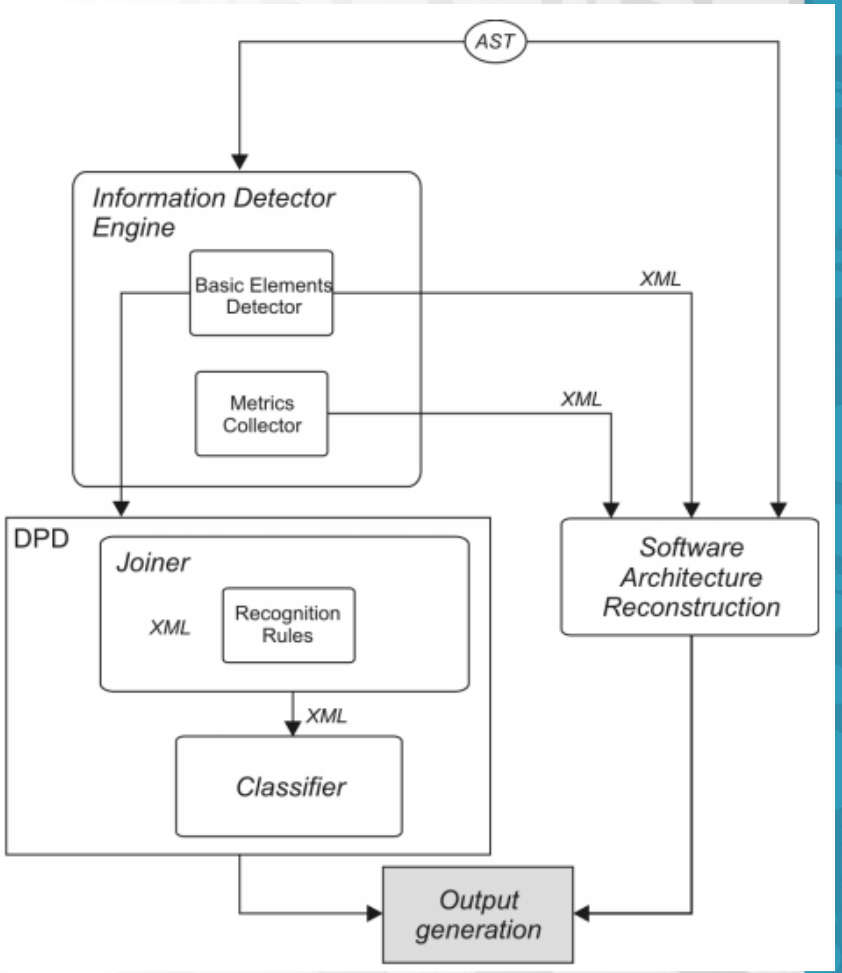


http://www.nicozazworka.com/tool-development/

Padrões de Código e Grime

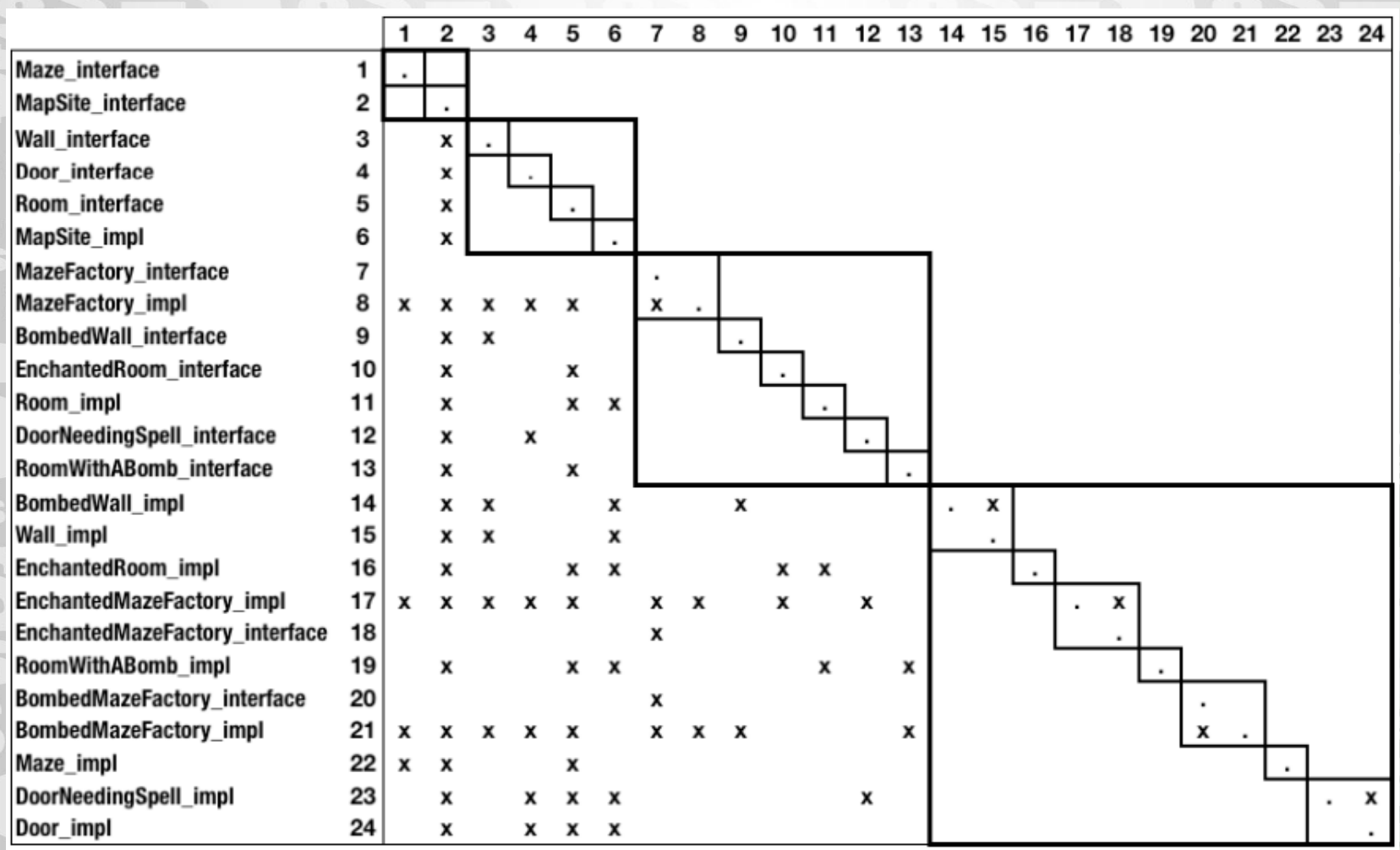
Padrões de Código e Grime

- Padrões de criação
- Padrões de Estrutura
- Padrões de Comportamento
- Padrões de Concorrência



http://essere.disco.unimib.it/reverse/files/MARPLE.pdf

Violações de Modularidade

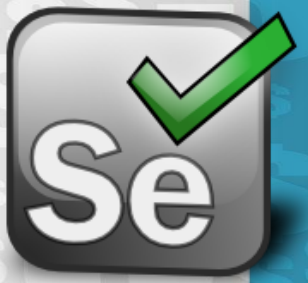


		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Maze_interface	1	.																							
MapSite_interface	2		.																						
Wall_interface	3		x	.																					
Door_interface	4		x		.																				
Room_interface	5		x			.																			
MapSite_impl	6		x				.																		
MazeFactory_interface	7							.																	
MazeFactory_impl	8	x	x	x	x	x		x	.																
BombedWall_interface	9		x	x						.															
EnchantedRoom_interface	10		x			x					.														
Room_impl	11		x			x	x					.													
DoorNeedingSpell_interface	12		x		x								.												
RoomWithABomb_interface	13		x			x								.											
BombedWall_impl	14		x	x			x			x					.	x									
Wall_impl	15		x	x			x									.									
EnchantedRoom_impl	16		x			x	x				x	x					.								
EnchantedMazeFactory_impl	17	x	x	x	x	x		x	x		x		x					.	x						
EnchantedMazeFactory_interface	18							x											.						
RoomWithABomb_impl	19		x			x	x					x		x						.					
BombedMazeFactory_interface	20							x													.				
BombedMazeFactory_impl	21	x	x	x	x	x		x	x	x				x							x	.			
Maze_impl	22	x	x			x																	.		
DoorNeedingSpell_impl	23		x		x	x	x						x											.	x
Door_impl	24		x		x	x	x																		.

https://www.cs.drexel.edu/~yfcai/CS680/Readings/Week9/ICSE11.ModularityViolation.pdf (CLIO)

Testes

- Testes de Unidade;
- Testes de Integração e
- Testes de Aceitação.



Coverage

TestAllPackages (Feb 13, 2012 9:52:22 AM)

Element	Coverage	Covered Lines	Missed Lines	Total Lines
commons-collections	80.7 %	11092	2646	13738
src	80.7 %	11092	2646	13738
org.apache.commons.collections	77.1 %	3991	1188	5179
org.apache.commons.collections.bag	66.9 %	234	116	350
org.apache.commons.collections.bidimap	91.2 %	964	93	1057
AbstractBidiMapDecorator.java	85.7 %	6	1	7
AbstractBidiMapDecorator	85.7 %	6	1	7
AbstractBidiMapDecorator(BidiMap)	100.0 %	2	0	2
getBidiMap()	100.0 %	1	0	1
getKey(Object)	100.0 %	1	0	1
inverseBidiMap()	0.0 %	0	1	1

http://www.eclemma.org/userdoc/images/coverageview.png

Medindo a Dívida Técnica

A dívida técnica DT, pode ser calculada pela seguinte fórmula:

$$DT = \sum_{i=1}^{n} f_i$$

Sendo f_i o custo de fazer a propriedade i ser válida em todos os pontos; calculada da seguinte forma:

$$f_i(P_1, P_2, \ldots, P_{m_i}) = \sum_{j=1}^{m_i} h_i(P_j)$$

em que P_j é um vetor que descreve cada ponto que está sendo avaliado

Medindo a Dívida Técnica

$$h_i = \begin{cases} 0, & se\ i\ não\ foi\ violada \\ x > 0, & caso\ contrário \end{cases}$$

representa o custo de fazer um ponto de avaliação (P_j) satisfazer a propriedade i.

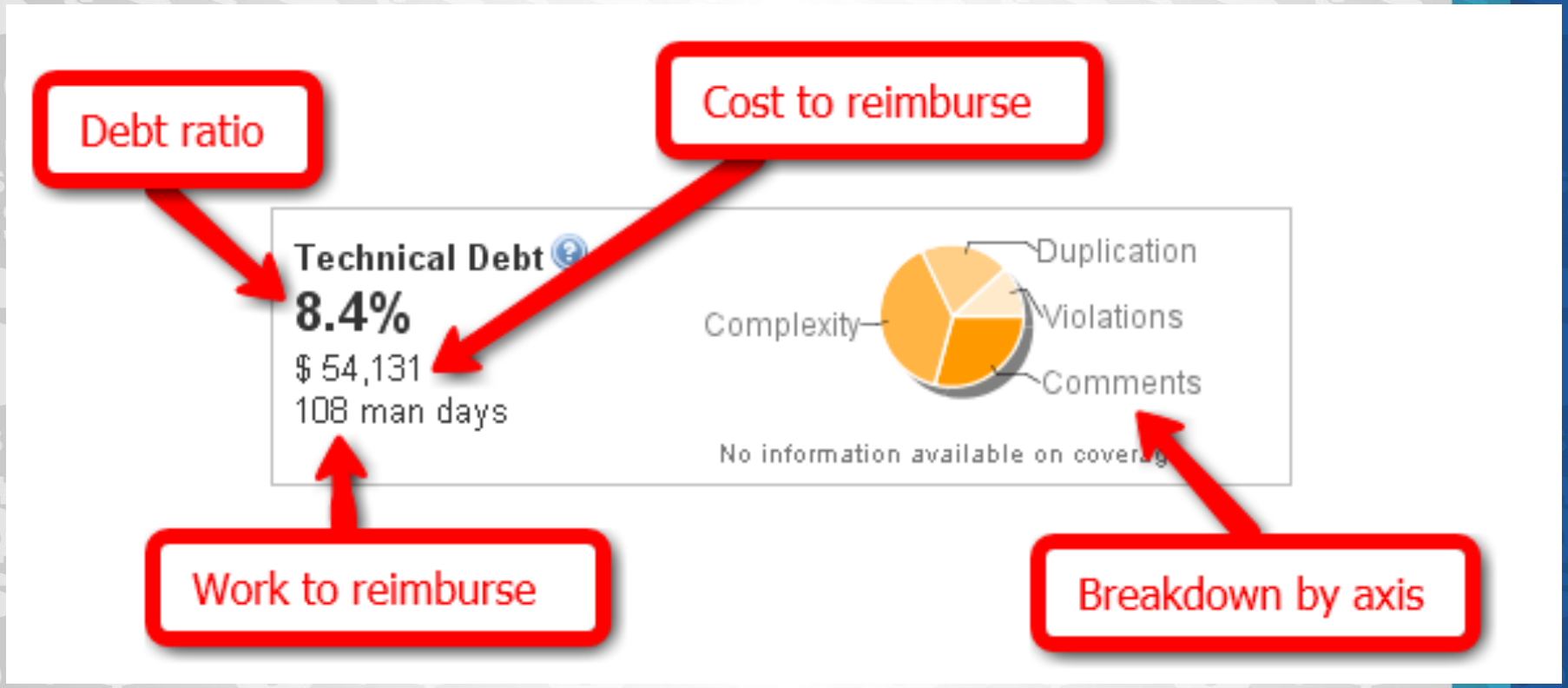
Essa função retorna um custo monetário, em horas ou um valor simbólico.

Armazenando a Dívida Técnica

ID	42
Data - Hora	01/06/13 - 11:30:34
Propriedade de Qualidade	Todo método deve estar coberto por pelo menos um teste de unidade.
Local	Classe ABC : Método XYZ
Custo de Remediação	2,5 horas
Prioridade para Pagamento	Alto

Variação do proposto em Measuring and Monitoring Technical Debit

sonarqube



http://docs.codehaus.org/display/SONAR/Technical+Debt+Plugin

Monitorando a Dívida Técnica

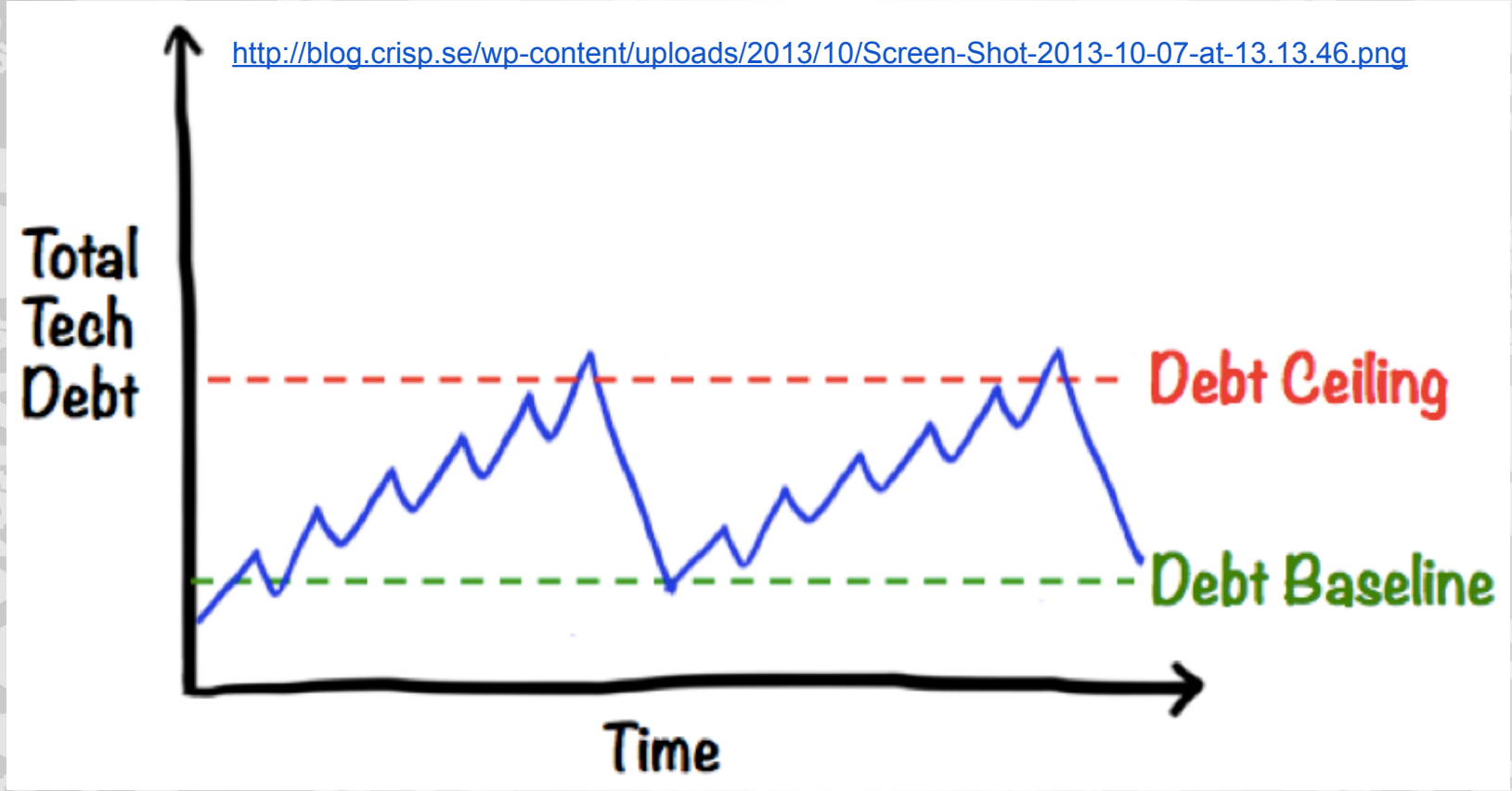
- Evitar a falência técnica/financeira do projeto;
- Controlar o custo de desenvolvimentos futuros e manutenção e
- Ajuda na tomada de decisões.

Arcabouço de Gerenciamento

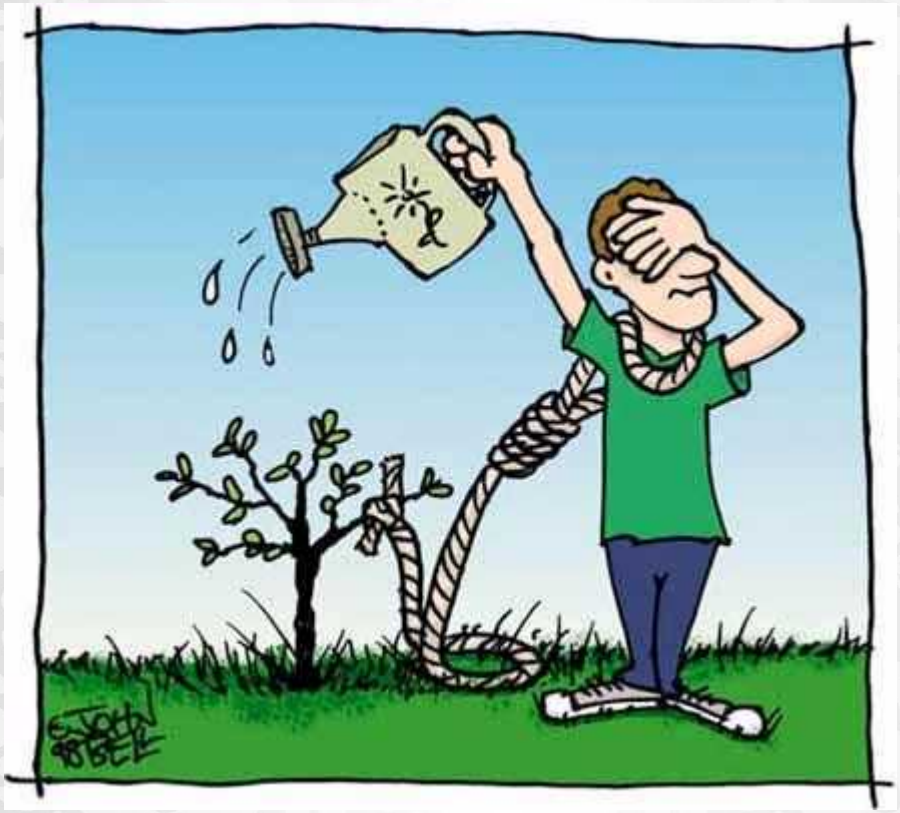
Modelo proposta por Carolyn Seaman

Tipos de Monitoramento

- Monitoramento Imediato e
- Monitoramento a Longo Prazo.



Como evitar a Dívida Técnica



http://3.bp.blogspot.com/_QfakWpKKrn8/S9M-KsFTcXI/AAAAAAAAAIo/yE7kn2DxXDw/s1600/dividas.jpg

Próximos Passos

- Integrar o Sonar Qube com alguma ferramenta para monitorar a dívida técnica
- Fazer uma interface para fácil integração entre as ferramentas de identificação
- Criar modelos para medir a dívida técnica

Referências

- Apresentação "Measuring and Monituring Technical Debt". Carolyn Seaman. 27 de Março de 2013 no IME-USP.
- Apresentação "Dívida Técnica". Alexandre Freire
- http://www.technicaldebt.umbc.edu
- https://code.google.com/p/findbugs-for-android
- **Measuring and Monitoring Technical Debt**, Carolyn Seaman and Yuepu Guo, *Advances in Computers*, Vol. 82, 25-46, 2011
- **Managing Technical Debt**, Steve McConnell, 10x Software Development, 1 de Novembro de 2007
- **Detecting Software Modularity Violations**, Sunny Wong, Yuanfang Cai, Miryung Kim e Michael Dalton
- http://en.wikipedia.org/wiki/Software_design_pattern
- http://grenoble.ime.usp.br/~gold/cursos/2013/movel/
- http://www-di.inf.puc-rio.br/~endler/courses/Mobile/
- http://essere.disco.unimib.it/reverse/files/MARPLE.pdf

http://www.todaletra.com.br/wp-content/uploads/2012/10/duvidas-300x3001.jpg