

Tabela 5.3. Análise do Tipo de Manutenção para os Cenários do MM [36].

Cenários	Descrição dos Cenários do MobileMedia	Caso 1	Caso 2
R1	Versão base do MobilePhoto [28].	-	-
R2	Inclusão do tratamento de exceção.	Aperfei.	Adição de requisito não-funcional.
R3	Novas funcionalidades adicionadas para contar e ordenar fotos pelo número de vezes que a mesma foi visualizada; e para editar a descrição das fotos.	Aperfei.	Adição de funcionalidade opcional; e adição de funcionalidade obrigatória.
R4	Nova funcionalidade adicionada para permitir que os usuários definam e vejam suas fotos favoritas.	Aperfei.	Adição de funcionalidade opcional.
R5	Nova funcionalidade adicionada para permitir que os usuários façam várias cópias de uma mesma foto; e aplicação de <i>Refactoring</i> nas regras do Controlador MVC.	Aperfei.	Adição de funcionalidade opcional; <i>Refactoring</i> .
R6	Nova funcionalidade adicionada para mandar fotos para outros usuários por SMS.	Aperfei.	Adição de funcionalidade opcional.
R7	Nova funcionalidade adicionada para controlar arquivos de música; e alteração do gerenciamento de fotos para uma funcionalidade alternativa.	Aperfei.	Alteração de uma funcionalidade obrigatória em duas alternativas.
R8	Nova funcionalidade adicionada para controlar arquivos de vídeo.	Aperfei.	Adição de funcionalidade alternativa.

* Aperfei.: Aperfeiçoamento.

5.4 Avaliação do Caso 2: Identificando aplicações e cenários *benchmark*

Esta seção apresenta a avaliação do segundo caso de utilização do BF, como mencionado anteriormente nas Seções 4.2 e 5.1. Neste caso, o mesmo é utilizado por Projetistas de Benchmarks, com o objetivo de guiar a inclusão ou alteração de aplicações e cenários a um Benchmark para o contexto de manutenção de software OA. Portanto, através do uso BF, os usuários podem avaliar as aplicações e cenários em questão, com relação à representatividade das características apresentadas por ambos os objetos de análise, aplicação e seus cenários de manutenção.

Sendo assim, o estudo realizado nesta seção tem como objetivo avaliar se o BF pode ser útil na identificação e/ou alteração de aplicações e cenários candidatos a *benchmark* de manutenção de software OA. Dessa forma, esta avaliação simula um Projetista de Benchmark utilizando o BF. O mesmo deve analisar se e como, HW, MM e seus respectivos cenários, precisam ser alterados para serem considerados *benchmarks* para estudos de manutenção de software OA. Como citado na Seção 5.1, as aplicações foram selecionadas, entre outros motivos, por pertencerem a diferentes domínios, terem diferentes características e serem alvos de estudos empíricos recentes [45][36]. Tais estudos disponibilizaram as listas de cenários utilizadas nesta avaliação.

Esses estudos têm como objetivo analisar a estabilidade arquitetural das aplicações quando estão sujeitas a cenários reais de manutenção. A escolha dos cenários foi baseada, exclusivamente, na experiência dos projetistas dos estudos, selecionando mudanças específicas que costumam afetar (positivamente e negativamente) a estabilidade da arquitetura de tais aplicações.

No caso do estudo descrito nesta seção, a utilização do BF como uma ferramenta para facilitar a identificação de *benchmarks* permite não só verificar se os candidatos a *benchmark* preenchem os requisitos para tal, assim como, avaliar indiretamente os estudos empíricos que elaboraram e adotaram tais objetos analisados pelo BF. Pois, ao classificar suas características específicas através dos critérios do BF, é possível identificar pontos de melhoria nas aplicações, cenários e, conseqüentemente, nos estudos que as adotaram.

Porém, nem sempre tais pontos de melhoria significam deficiências no estudo. Dependendo dos seus objetivos, essas possíveis carências podem ter sido planejadas como as outras decisões experimentais. Em tais casos é preciso investigar detalhadamente os planos experimentais para descobrir a motivação dessas decisões. Uma prova disto é que ambos os estudos empíricos analisados apresentam conclusões bastante interessantes, independentemente dos resultados desta avaliação, que são apresentados a seguir.

A seção a seguir mostra os resultados da avaliação divididos por aplicação (produto), que utiliza os critérios da Seção 4.3 e cenários (cenários de manutenção), que utilizam os critérios da Seção 4.4.

5.5 Análise dos Resultados para o Caso 2: Identificando aplicações e cenários *benchmark*

Esta seção apresenta os resultados obtidos na utilização dos critérios do BF para avaliar as aplicações e suas versões, candidatas a *benchmark*, na presença de vários tipos de mudança. Os resultados da avaliação têm como principal objetivo, investigar a adequação e representatividade das versões dos sistemas para servirem de aplicações *benchmark* em futuros estudos empíricos, de acordo com os critérios do BF (Capítulo 4). A avaliação da representatividade dos sistemas selecionados foi dividida em duas partes, equivalentes aos critérios do BF:

- Análise das características dos sistemas em termos dos critérios do produto (Seção 5.5.1)
- Análise das características dos cenários de manutenção existentes de acordo com os critérios dos cenários de mudança (Seção 5.5.2).

Apesar das aplicações analisadas possuírem vários pontos e boa representatividade em algumas circunstâncias, os resultados deste estudo indicam que ambas as aplicações possuem algumas lacunas com relação à diversidade de suas características. Tais pontos poderiam ter sido identificados e evitados facilmente, caso o BF tivesse sido utilizado em seu planejamento.

Contudo, os resultados indicam como as aplicações e seus cenários podem ser adaptados, para que passem a atender a tais características que ainda não estão presentes. Esta informação auxilia na replicação de novos estudos empíricos que se baseiem nos estudos analisados por este

trabalho, e bem como, facilitando a geração de aplicações *benchmark*. Além da análise dos resultados, as seguintes seções também introduzem algumas discussões reflexivas sobre as lições aprendidas com a utilização do BF.

Outra observação importante é que apenas parte dos critérios do BF foi considerada nesta avaliação, pois nem todos estão diretamente relacionados com o objetivo principal dos estudos, que é analisar a modularidade do sistema no nível arquitetural. Esta e outras limitações desta avaliação estão descritas na Seção 5.6.

As seções a seguir descrevem as duas partes desta avaliação.

5.5.1 Avaliando o Produto

A fim de investigar a possibilidade de HW e MM serem consideradas aplicações *benchmark*, o seguinte subconjunto de critérios do BF foi utilizado para avaliar as características específicas de cada uma das aplicações: Classificação de Interesses Transversais, Classificação de Interação e Composição de Interesses, Escopo de Interesses Transversais e Construções para Linguagens OA.

Classificação de Interesses Transversais

A Tabela 5.4 mostra que o HW não abrange a todos os tipos de interesses transversais descritos nos critérios do BF. O único tipo de interesse transversal que não está presente é o de interesses transversais funcionais. Isso é um fator bastante relevante, pois, é possível concluir que as regras de negócio existentes no HW, não forçam situações de instabilidades arquiteturais, como sistemas que possuem regras que contenham interesses transversais funcionais.

Uma abordagem interessante para solucionar este problema seria a criação de um cenário de manutenção para implementar tal tipo de interesse transversal. Porém, observando a descrição dos cenários (segunda coluna) na Tabela 5.2, tal característica não está presente em nenhum dos cenários elaborados pelo estudo original [45].

Já no caso do MM, de acordo com a Tabela 5.4, a aplicação possui sem exceção, todos os tipos de interesses transversais descritos no BF, como, Funcional e Não-Funcional, Homogêneo e Heterogêneo, e Intra-Componente e Inter-Componente.

Tabela 5.4. Classificação dos Tipos de Interesses Transversais.

	Funcional	Não-Funcional	Homogêneo	Heterogêneo	Intra-Componente	Inter-Componente
HW	-	X	X	X	X	X
MM	X	X	X	X	X	X

Classificação de Interação e Composição entre Interesses

Observando a Tabela 5.5, percebe-se que HW e MM possuem boa representatividade em termos de diferentes tipos de interação e composição entre interesses, pois ambos atendem a todos os critérios definidos no BF, como Baseado em Invocação, Entrelaçamento no Nível de Componentes, Entrelaçamento no Nível de Operações e Sobreposição.

Tabela 5.5. Classificação dos Tipos de Interação e Composição de Interesses de Interesses Transversais.

	Baseado em Invocação	Entrelaçamento no Nível de Componentes	Entrelaçamento no Nível de Operações	Sobreposição
HW	X	X	X	X
MM	X	X	X	X

Escopo de Interesses Transversais

Tanto no HW quanto no MM, todos os interesses transversais se manifestam durante as atividades de projeto e implementação. Por isso, seria interessante que fossem adicionados cenários que apresentassem manifestação de interesses transversais em diferentes atividades do desenvolvimento dos softwares, como por exemplo, durante as atividades de engenharia de requisitos. Dessa forma, seria possível avaliar de forma mais abrangente como a arquitetura do software seria impactada pela manifestação de interesses transversais ocorrendo em diferentes atividades.

Construções para Linguagens OA

As versões analisadas de HW e MM foram implementadas utilizando AspectJ. Portanto, a avaliação se baseou nas construções desta linguagem. Como mostra a Tabela 5.6, foram investigados os artefatos gerados pela primeira e última versão de cada aplicação, ou seja, R1 e R9 para o HW, e R2 e R8 para o MM. Neste caso, a versão considerada como inicial não é a R1, pois em tal versão só existe o código original OO. O objetivo desta avaliação é verificar a presença de diferentes construções no produto final e verificar se os cenários foram ou não responsáveis por adicioná-los. Isto auxilia a decisão se as aplicações finais e seus respectivos cenários podem ou não ser considerados *benchmarks*.

As versões finais de HW (R9) e MM (R8) apresentam uma vasta variedade de construções de linguagem OA, incluindo:

- Declaração de intertipos, como por exemplo, *declare parents*, *declare soft*, *declare precedence* e declaração de novos membros;
- Aspectos abstratos e privilegiados;
- Vários tipos de designadores de pontos de atuação, como por exemplo, chamada e execução de métodos, além de pontos de atuação abstratos;
- Diferentes tipos de adendos, como por exemplo, *before*, *around* (com e sem *proceed*), *after*, *after returning* e *after throwing*;

Porém, nem todos os tipos de construções são encontrados em ambas as aplicações, como por exemplo, apenas o HW apresenta o uso de modificador do controle de fluxo *cflow*, e já o *cflowbelow* não é encontrado em nenhuma das aplicações. Outros exemplos são as construções *within* e *withincode*, onde apenas a primeira está presente no HW e apenas a segunda no MM.

Ainda sim, mesmo com essas ausências existem vários tipos de mecanismos de programação de AspectJ presentes no HW e MM, ou seja, sob esse ponto de vistas elas são bastante representativas. No entanto, quando essas construções são avaliadas em termos de como elas afetam o código fonte, tais como, construções homogêneas e heterogêneas, é possível verificar algumas questões interessantes, como as exibidas na Tabela 5.6.

Tabela 5.6. Construções Homogêneas e Heterogêneas de Linguagens OA.

Tipos de Construções			Aplicações		HW		MM	
			NOO	LOC	R1	R9	R2	R8
Ponto de Atuação	Homogêneo	NOO	7	15	0	0	0	0
		LOC	10	28	0	0	0	0
	Heterogêneo	NOO	5	28	19	70	19	70
		LOC	6	47	38	147	38	147
Adendo	Homogêneo	NOO	9	17	0	0	0	0
		LOC	60	134	0	0	0	0
	Heterogêneo	NOO	6	29	19	66	19	66
		LOC	51	283	129	608	129	608
Intertipo	Homogêneo	NOO	10	14	1	1	1	1
		LOC	19	28	1	1	1	1
	Heterogêneo	NOO	6	16	21	84	21	84
		LOC	18	30	21	282	21	282

NOO: Número de Ocorrências (Number of Occurrences) e LOC: Linhas de Código (Lines of Code).

Estes resultados foram coletados em termos de Número de Ocorrências (NOO) e da quantidade de Linha de Código (LOC) associada a cada uma dessas ocorrências. Com base nos dados apresentados, pode-se concluir que, sob esse aspecto, o MM não é representativo o suficiente, pois as suas versões, inicial e final, não possuem construções homogêneas, existe apenas uma ocorrência de intertipo e nenhuma de ponto de atuação e adendo. Por outro lado, o HW possui uma boa diversidade de construções homogêneas e heterogêneas.

Os resultados, no caso do MM, indicam que o código fonte relacionado à implementação dos aspectos não está sendo reusado, uma vez que todas as construções OA (exceto uma) são heterogêneas, afetando apenas um único ponto de junção para cada ponto de atuação (Seção 4.3.2). O tamanho final do código fonte também é diretamente influenciado, isto acontece devido ao uso exclusivo de construções heterogêneas, que reduzem a vantagem do reuso e, conseqüentemente, aumentam a repetição de código fonte e dificultam a manutenção do software.

5.5.2 Avaliando os Cenários

Os cenários das aplicações HW e MM, apresentados na segunda coluna das Tabelas 5.2 e 5.3, são avaliados nesta seção com objetivo de verificar se possuem as características necessárias a um cenário *benchmark*. Esses cenários foram submetidos à avaliação de um subconjunto de critérios do BF, responsável por investigar as características específicas dos cenários candidatos a benchmark. Tal subconjunto é composto por: Tipo de Mudança e Nível de Mudança.

Tipo de Mudança

A Tabela 5.7 apresenta os resultados da classificação dos tipos de mudança presentes em cada um dos cenários de HW e MM de acordo com os critérios presentes no BF. Nesta avaliação tanto o objetivo quanto a natureza da mudança são considerados para avaliar a representatividade dos cenários implementados.

Os resultados mostram que os cenários de HW e MM são pouco representativos com relação à diversidade dos objetivos da mudança, pois, ambas as aplicações possuem um único tipo, Aperfeiçoamento, que é equivalente a apenas 33% das possíveis categorias listadas no BF. Porém, como mostra a Figura 5.1, as outras duas categorias existentes nos critérios, Corretiva e Adaptativa, representam respectivamente 17% e 18% das mudanças reais que acontecem em sistemas de software. Conseqüentemente, a categoria de Aperfeiçoamento, que está presente nas aplicações, é responsável por 65% do total, segundo estudos realizados sobre como é a real distribuição dos tipos de manutenção em diferentes softwares [66][76][97].

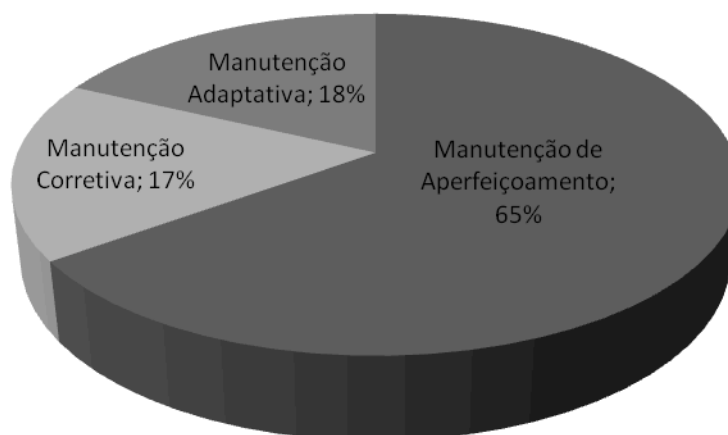


Figura 5.1. Representação Real da Distribuição dos Tipos de Manutenção [97].

Outro ponto que não é coberto satisfatoriamente é relacionado à natureza das mudanças, mais especificamente ao comportamento do sistema na presença das mudanças. Neste caso, os cenários de HW e MM apresentam características opostas, enquanto o primeiro apresenta apenas 2 cenários com alteração de comportamento (R2 e R8) e outros 7 com preservação, o segundo possui 6 cenários com alteração e apenas 1 com preservação de comportamento (R2). Dessa forma pode-se concluir que os cenários do HW são pouco representativos em mudanças que alterem o comportamento da aplicação, e por outro lado, os cenários do MM são pouco representativos em mudanças que preservem o comportamento da aplicação. Logo, o projeto de um novo estudo empírico relacionado a estes poderia considerar o uso de ambas as aplicações, já que elas possuem características distintas e complementares.

Os resultados também são negativos quando analisados em termos da representatividade dos cenários com relação ao total de possíveis combinações entre os tipos de objetivo e natureza das mudanças. Por exemplo, todos os 16 cenários analisados se concentram em apenas duas possíveis classificações: o primeiro grupo possui 8 cenários com o objetivo de manutenção de Aperfeiçoamento e mudanças que preservam o comportamento da aplicação, enquanto o segundo grupo, também com 8 cenários, possui o objetivo de manutenção de Aperfeiçoamento e as mudanças alteram o comportamento da aplicação.

Portanto, de um total de 6 possíveis categorias, os cenários analisados atendem a apenas duas delas, conseqüentemente, não atendem as outras 4 possíveis categorias (colunas 2, 3, 4 e 5 na Tabela 5.7), equivalente a 66% do total de categorias.

Tabela 5.7. Classificação dos Tipos de Mudança (Objetivo e Natureza) Para Cada Cenário de HW e MM.

Cenários	Tipo da Mudança					
	Corretiva		Adaptativa		Aperfeiçoamento	
	Preserva o Comportamento	Altera o Comportamento	Preserva o Comportamento	Altera o Comportamento	Preserva o Comportamento	Altera o Comportamento
HW						
R1					X	
R2						X
R3					X	
R4					X	
R5					X	
R6					X	
R7					X	
R8						X
R9					X	
MM						
R1	-	-	-	-	-	-
R2					X	
R3						X
R4						X
R5						X
R6						X
R7						X
R8						X

Nível da Mudança

A Tabela 5.8 mostra os resultados de todas as mudanças estruturais ocorridas nas aplicações durante a implementação dos primeiros cenários. Os valores foram coletados durante a transição de R1 para R2 no HW e R2 para R3 no MM. Dessa forma, pode-se observar exatamente o impacto estrutural causado a cada elemento da aplicação (em termos de adição, alteração ou remoção), pelo desenvolvimento de um cenário específico, neste caso, R2 para o HW e R3 para o MM.

Tabela 5.8. Mudanças no Nível de Implementação.

Unidade de Implementação		Natureza da Mudança	Adicionado	Alterado	Removido
HW	OA	Aspecto	3	5	0
		Ponto de Atuação	6	4	0
		Adendo	5	4	0
		Intertipo	2	0	0
		Método	7	0	0
		Atributo	25	0	0
	OO	Classe	23	2	19
		Método	28	0	22
		Atributo	9	5	5
MM	OA	Aspecto	1	0	0
		Ponto de Atuação	6	4	0
		Adendo	6	3	0
		Intertipo	4	4	0
		Método	2	0	0
		Atributo	2	0	0
	OO	Classe	1	6	0
		Método	8	10	0
		Atributo	7	4	0

Os resultados apresentam, para ambas as aplicações, uma boa representatividade em termos de elementos adicionados, pois são adicionados todos os tipos presentes na classificação. Com relação aos elementos alterados, existe uma representatividade intermediária, uma vez que

alguns elementos não possuem alteração, como por exemplo, métodos e atributos OA para as duas aplicações. E por último, pode-se observar uma fraca representatividade de elementos removidos, pois, apenas o HW apresenta dois tipos de elementos removidos, que apesar de serem muitos são exclusivamente classes e métodos OO.

Esta avaliação apresentou os resultados referentes a uma única transição, mas para avaliar detalhadamente todos os cenários seria interessante avaliar a transição de todos eles, para que seja possível identificar pontos de melhoria em cada um deles.

5.6 Limitações da Avaliação

Mesmo tendo realizado diferentes avaliações (Seção 5.2 a 5.4) que confirmaram os supostos benefícios do BF através de uma investigação preliminar da sua utilidade, os procedimentos realizados tiveram algumas limitações. Tais limitações irão contribuir para futuros estudos utilizando outras aplicações candidatas a *benchmark*, e diferentes procedimentos experimentais como alvo.

Nem todos os elementos do BF foram avaliados detalhadamente, por exemplo, ainda não foi feita uma avaliação sobre como a utilização do BF pode ser efetiva para auxiliar a geração de *cookbooks* (saída ① do BF na Figura 4.1) para manutenção de sistemas de software OA [22]. Da mesma forma que nem todos os critérios do BF foram exaustivamente empregados nas avaliações apresentadas neste capítulo, sendo mais um ponto a ser focado por trabalhos futuros, assim como o refinamento de tais critérios.

Outro ponto importante são deficiências identificadas durante a avaliação de comparação dos planos de estudos empíricos (Seção 5.3). O plano do especialista continha algumas informações que não estavam presentes no BF, tais como, um conjunto de métricas para avaliar as instabilidades arquiteturais, que podem gerar um novo módulo do BF através da sua extensão. Tais aperfeiçoamentos incluem também a criação de um novo módulo, para auxiliar uma avaliação mais quantitativa da eficácia e completude que o BF possui ao atender os seus objetivos, que é uma dificuldade freqüente na área de Engenharia de Software.

Outro fator que poderia ser definido como uma limitação é o fato dos orientadores deste trabalho terem se envolvido na elaboração dos estudos que foram selecionados como alvo para participarem desta avaliação [36][45]. Porém, o autor deste trabalho não participou em nenhum momento ou foi influenciado por tais estudos durante a elaboração do BF. Assim como, grande

parte dos critérios existentes no BF foram selecionados através de trabalhos de autoria independente, sem que houvesse qualquer possibilidade influência.

5.7 Considerações Finais

Esta seção descreve resumidamente as informações contidas neste capítulo. Os resultados das avaliações do BF comprovam os seus supostos benefícios, mostrando que o mesmo pode ser útil para atender a diferentes objetivos, tipos de usuários e domínios de aplicação, dentro do contexto de manutenção de software OA. Porém, apesar dos benefícios comprovados, o mesmo ainda possui algumas limitações, como mencionado na seção anterior.

Os dois tipos de avaliações: (i) elaboração novos experimentos com o BF (Seção 5.2) e, (ii) identificação de aplicações e cenários *benchmark* (Seção 5.4) indicam que a utilização do BF é vantajosa, inclusive para especialistas na área, sejam projetistas de estudos empíricos ou projetistas de benchmarks. Uma vez que as avaliações mostraram que o BF é capaz de suprir algumas deficiências dos especialistas, porém em nenhum momento, substitui a experiência dos mesmos.