

Fábrica de Software: da Definição às Lições Aprendidas

Breno Gustavo Soares da Costa
Carlos A. Rego Andrade
Carlos Roberto da Silva Júnior
Clarissa César Borba
Eduardo Santana de Almeida
Fábio de Carvalho Dias Ferreira
Luiz Eduardo Cunha Leite
Mônica Priscila R. França
Taciana Amorim Vanderley
Vanessa Bittencourt X. de Moura
Vivianne da Nóbrega Medeiros

fabrica3@cin.ufpe.br

Universidade Federal de Pernambuco
Centro de Informática

Resumo

Pesquisas e esforços envolvendo fábricas de software tem sido apresentado ao longo dos anos na literatura. Entretanto, ainda existe uma carência de relatos de experiências envolvendo as etapas necessárias para sua construção. Assim, este artigo apresenta experiências com uma fábrica de software descrevendo desde a fase de definição, até as lições aprendidas durante todo o processo, oferecendo uma contribuição significativa no seguimento de fábricas de software.

Palavras-chaves: Engenharia de Software, Fábrica de Software.

1. Introdução

A performance cada vez mais otimizada das fábricas industriais clássicas, a consolidação das técnicas de engenharia de software, juntamente com o refinamento dos ambientes de desenvolvimento e o surgimento de novos ambientes de projeto e suporte integrados têm feito com que, cada vez mais, esforços sejam despendidos no sentido de realizar o conceito de fábrica de software. Esse conceito simboliza uma desejada mudança de paradigma da produção de software focada no trabalho intensivo, para um estilo mais focado no capital, onde investimentos substanciais podem ser feitos sob um nível de risco aceitável.

As fábricas clássicas, onde as pessoas atuam como máquinas na realização de tarefas predeterminadas, não é um modelo nem desejável, nem correto para fábrica de software. No contexto do software, a analogia com a fábrica pode ser aplicada apenas aos objetivos da produção baseada no estilo industrial, e não na sua implementação. A manufatura de software envolve pouca ou nenhuma produção tradicional: todo sistema é único; apenas partes individuais podem aparecer repetidamente em mais de um sistema [Fernström 1992].

Uma importante característica de uma fábrica de software é a importância que ela dá a informação acumulada de vários projetos. Esta informação pode adquirir várias formas, incluindo elementos reutilizáveis (código, projeto e documentação), medidas de

performance, relatórios da aplicação de técnicas específicas e processos de desenvolvimento.

Neste trabalho, apresentamos uma análise pormenorizada sobre pontos positivos e negativos encontrados durante o processo de elaboração e gestão de uma fábrica de software. Essa fábrica, denominada Fábrica III, foi criada com o intuito de validar o conceito de fábrica de software, passando pelas atividades que vão desde a criação e definição da forma gestão da fábrica, até o desenvolvimento de aplicações pela mesma.

O artigo está organizado em quatro seções. A primeira seção contém esta introdução. A seção 2 descreve o processo de criação da fábrica propriamente dito, sendo dividido em sete subseções que contém, respectivamente, a definição, o planejamento, o projeto utilizado, a instanciação, a execução e, por fim, a análise dos resultados juntamente com as lições aprendidas. A seção 3 apresenta os trabalhos relacionados e, finalmente, a seção 4 apresenta as conclusões e perspectivas de trabalhos futuros.

2. Fábrica de software: da Definição às Lições Aprendidas

A criação de uma fábrica de software pode ser dividida em cinco etapas: a **definição**, o **planejamento**, a **execução**, a **análise** e o **empacotamento** do processo. A definição é a primeira etapa, onde se define a fábrica em função do contexto, objetivo, estratégia, organização e, por fim, melhoria de implementação. O planejamento envolve a descrição do perfil dos participantes, dos instrumentos, do processo de execução e uma avaliação dos problemas que podem vir a ocorrer durante sua execução. A execução consiste na realização do projeto pelos participantes, utilizando os instrumentos e o processo definidos anteriormente. A fase de análise consiste na organização dos resultados gerados pelos participantes durante a execução e a inferência sobre estes resultados. Finalmente, o empacotamento consiste na organização e armazenamento dos artefatos previamente construídos.

As etapas de definição e planejamento, apresentada nas próximas seções, são descritas em tempo verbal futuro, simbolizando a precedência do plano à execução.

2.1 Definição da Fábrica de Software

No estudo realizado sobre fábricas de software [Aaen, 1997], os autores definiram uma abordagem para comparação de fábricas de software, de modo que se permite realizar o entendimento e o contexto no qual a fábrica foi concebida.

Assim, utilizando a abordagem de Aaen, temos:

2.1.1 Contexto

A motivação para o estabelecimento da Fábrica III é definir um processo eficiente de desenvolvimento de software baseado em componentes, com métodos, técnicas e ferramentas, tendo produtos de software desenvolvidos num ambiente acadêmico, através do Centro de Informática (CIn), da Universidade Federal de Pernambuco, Brasil.

2.1.2 Objetivo

Atualmente, torna-se extremamente necessário que mudanças sejam feitas na estrutura atual no qual o software é produzido. Empresas necessitam incrementar a produtividade e a

qualidade dos seus processos e produtos, assim como reduzir custos. Por conseguinte, uma proposta de solução deve considerar três importantes pontos:

- i. Melhorar a eficiência dos processos;*
- ii. Reduzir a quantidade de re-trabalho; e*
- iii. Considerar a reutilização de artefatos previamente construídos.*

2.1.3 Estratégia

A estratégia que deve ser baseada sob a premissa de melhoria contínua, consiste de quatro elementos chaves: o processo, o progresso, o produto e o participante. O processo descreverá os *guidelines* que devem ser seguidos durante o desenvolvimento. O progresso apresentará avaliações processuais, dando um *feedback* contínuo e incremental das tarefas realizadas. O produto será desenvolvido dentro de uma abordagem baseada em componentes, objetivando uma maior reutilização e redução nos custos e no tempo de desenvolvimento. O participante será o profissional definido para seguir o processo a ser adotado.

2.1.4 Organização

A organização da Fábrica III será definida por um processo que atende à estrutura comercial, técnica e gerencial da empresa.

A empresa possuirá uma metodologia de desenvolvimento de software composto de quatro fases: vendas, planejamento e gerenciamento, desenvolvimento de componentes e, por fim, testes e validação.

2.1.5 Melhoria de Implementação

O uso efetivo da reutilização requer uma abordagem incremental, onde a necessidade de disseminação do conhecimento é constante, além do alto gerenciamento tornar-se um fator crucial [Frakes, 1995].

Deste modo, os elementos chaves visando gerenciar o processo de melhoria contínua será composto das seguintes etapas: alto gerenciamento, identificar resistências na fábrica, quanto aos processos propostos e, por fim, disseminar o conhecimento entre os participantes.

2.2 Planejamento da Fábrica de Software

A fase de planejamento da fábrica deve considerar os seguintes aspectos como primordiais:

- i. Definir os perfis funcionais que realizarão as atividades de cada processo e em quais perfis cada participante deverá estar.*
- ii. Definir a metodologia de desenvolvimento de software a ser utilizada.*
- iii. Definir o plano de processos, descrevendo as atividades e relacionando-as com os artefatos e perfis funcionais que a executarão.*
- iv. Definir todo o material de instrumentação utilizado pelos participantes.*

Perfis: os participantes da fábrica serão alocados de acordo com os sete perfis, a saber:

- *Gerente de negócios:* responsável pela prospecção do mercado e venda dos serviços.
- *Gerente de Projeto:* responsável pelo gerenciamento das atividades de desenvolvimento, devendo dimensionar e alocar os recursos necessários, além de interagir com o cliente e o gerente de negócio.

- *Analista de Sistemas*: responsável pelo levantamento de requisitos, análise, processo de desenvolvimento e definição da arquitetura de desenvolvimento, assim como sua documentação.
- *Analista de Qualidade*: responsável pela definição e validação da qualidade e correção de todo o processo utilizado pela fábrica.
- *Engenheiro de Software*: responsável pelo desenvolvimento de software, em conformidade com as especificações e documentação, seguindo o processo de desenvolvimento definido.
- *Engenheiro de Testes*: responsável pelo desenvolvimento, validação e execução de testes de software com o intuito de assegurar a qualidade e correção do software produzido.
- *Líder de Equipe*: responsável pela coordenação e execução das atividades de desenvolvimento de software.

Metodologia de desenvolvimento: a metodologia de desenvolvimento será baseada no Rational Unified Process (RUP) [Jacobson, 2001] e no PMBOK [Cleland, 2000] será dividida nas seguintes fases:

- *Comercial*: é a fase de definição do projeto, de acordo com as necessidades levantadas junto ao cliente.
- *Planejamento e Gerenciamento*: é a fase da elaboração do plano do projeto (plano de trabalho, riscos, acompanhamento e controle) e execução das atividades recorrentes de acompanhamento e controle.
- *Desenvolvimento de Componentes*: é a fase de definição do problema e da especificação, projeto e implementação dos componentes. Esta fase segue uma abordagem definida em [Almeida, 2003].
- *Testes e Validação*: é a fase dos testes e da validação dos artefatos previamente construídos.

Plano de processos: o plano de processos irá detalhar cada fase da metodologia, mostrando o fluxo de atividades que a define. A Figura 1 mostra o plano de processo a ser adotado pela fábrica.

Cada atividade descrita descreve as informações de quais são os seus objetivos, quais são os insumos necessários à sua execução (entradas), a seqüência de passos a serem executados, os resultados gerados por essa atividade (saídas) e os perfis responsáveis pela sua execução. A Figura 2 mostra uma descrição da atividade *Levantar necessidades do cliente* na notação que foi utilizada no Plano de Processo.

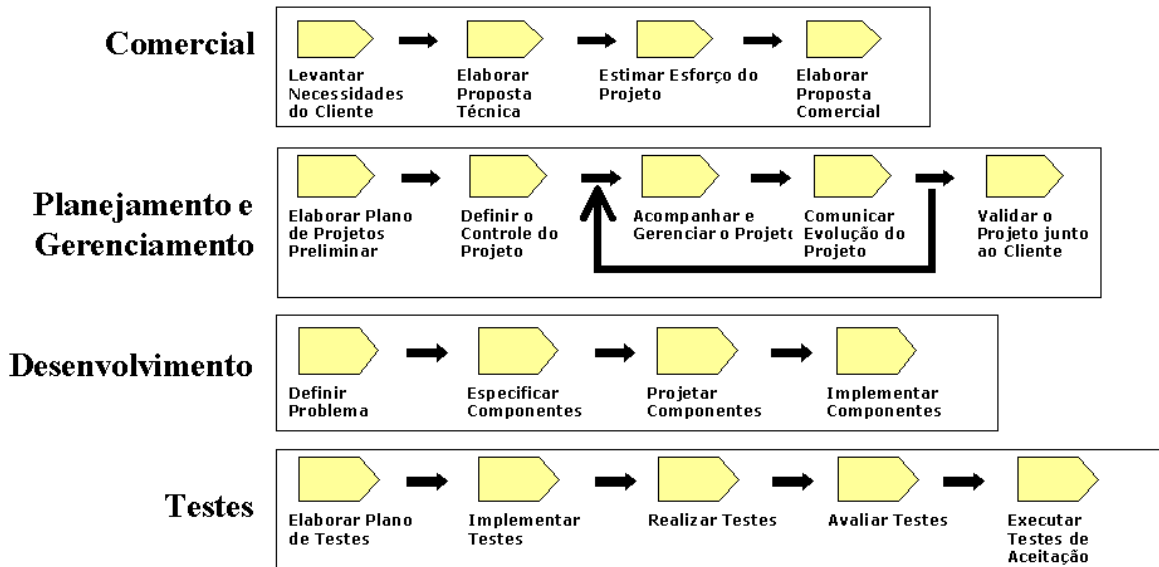


Figura 1. Plano de Processo utilizado pela Fábrica.

Fase: Comercial	Atividade: Levantar necessidades do cliente
Objetivos: Levantar e descrever as necessidades do cliente, identificando as soluções.	
Entradas: • Documentos apresentados pelo cliente	Saídas: • Ata de reunião
Passos: 1. Identificar usuários e clientes do sistema 2. Identificar requisitos funcionais e não-funcionais	
Responsável: Gerente de negócios	
Referências: “-”	

Identificar Usuários e Clientes do Sistema

Nesse passo, são identificados os usuários e clientes do sistema, que representarão os atores no diagrama de requisitos. É importante especificar quais são as responsabilidades e papéis dos clientes e usuários para que o documento também sirva como entrada para o entendimento do sistema.

Identificar Requisitos Funcionais e Não-Funcionais

Nesse passo, serão identificados os casos de uso, com a lista de passos do fluxo de eventos principal. Os casos de uso serão necessários para a estimativa de esforço na atividade subsequente.

Figura 2. Atividade Levantar necessidades do cliente.

Treinamento: o treinamento dos participantes da fábrica será realizado no próprio ambiente acadêmico e será dividido entre os setores de vendas, planejamento e gerenciamento, desenvolvimento de componentes e, por fim, testes e validação.

Projeto Piloto: antes da realização do primeiro projeto real, será definido um projeto piloto a fim de identificar possíveis inconsistências durante o processo e evitar o seu posterior impacto na fase de produção.

Instrumentação: cada participante deverá atuar em um dos perfis definidos anteriormente. Todos os participantes receberão um questionário para avaliação do processo utilizado pela fábrica, assim como um contabilizador de tempo e atividades realizadas. As tecnologias a serem empregadas pela fábrica serão definidas por categorias:

- i. Ferramentas de Desenvolvimento e Modelagem.*
- ii. Ferramentas para relatar Bugs durante o processo de desenvolvimento.*
- iii. Ferramentas de Gerenciamento de Projetos.*
- iv. Ferramentas para comunicação entre os participantes e disseminação do conhecimento.*
- iv. Sistema gerenciador de banco de dados.*

Análise Qualitativa: tem o objetivo de avaliar as dificuldades encontradas durante a execução do processo e a qualidade do treinamento realizado. Essa análise será realizada através do questionário utilizado pelos participantes desde o seu ingresso na fábrica, até o seu primeiro projeto realizado. Esse questionário será de suma importância, uma vez que através de sua utilização será possível avaliar as dificuldades encontradas pelos participantes com a utilização do processo, avaliar a utilização do treinamento disponibilizado, assim como permitir realizar melhorias nesses artefatos, visando à melhoria contínua.

Seleção Randômica: esta técnica pode ser utilizada na seleção dos perfis dos participantes, porém, os mesmos serão alocados de acordo com seus conhecimentos pré-definidos.

2.3 Projeto Piloto

2.3.1 Contexto

Com o objetivo de validar e ajustar a fábrica definida, um projeto piloto foi realizado. Esse projeto piloto não deveria ser complexo, de modo que o mais importante seria o acultramento dos participantes nos processos, atividades, artefatos e tecnologias.

Neste sentido, o projeto piloto foi à implementação de um sistema de alocação de disciplinas, professores, salas e horários, para auxiliar os funcionários da secretaria de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco.

2.3.2 Execução

A execução do piloto aconteceu sem maiores problemas e o sistema foi implementado com o escopo requisitado e o prazo previsto.

Essa primeira execução da fábrica foi de extrema importância para que os participantes ganhassem experiência na metodologia definida, assim como a verificação da adequação das pessoas em relação aos perfis funcionais e para que todo o processo operacional de uma fábrica de software pudesse ser experimentado pelos participantes.

2.3.3 Análise

Durante a execução do projeto piloto, alguns pontos merecem ser analisados:

i. Sobrecarga de Trabalho. Durante a execução do projeto, houve sobrecarga de trabalho relacionado à equipe de desenvolvimento, que efetuou um trabalho mais dispendioso que outros setores, como, por exemplo, comercial, testes e gerência de projetos. Este fato ocorreu, principalmente, pela baixa interseção dos horários de disponibilidade entre os membros da fábrica, levando, muitas vezes, a falhas de comunicação.

ii. Alocação de Perfis. Alguns participantes não estavam bem alocados em relação ao perfil funcional que se encontravam causando, deste modo, alguns atrasos de cronograma.

iii. Instrumentação. A ferramenta CASE adotada inicialmente para modelagem do sistema não estava atendendo da forma esperada, uma vez que não permitia o trabalho colaborativo, ocasionado a ociosidade de alguns participantes.

Em virtude destes fatores, e pela própria execução do piloto, a fábrica sofreu alguns ajustes:

i. Realocação de Perfis. Foi realizado a realocação de três participantes em relação aos perfis funcionais, para melhor adequação aos futuros projetos.

ii. Ajustes de Métricas. O multiplicador de Ponto de Caso de Uso Ajustado (PCUA) para Homem/Hora (HH), proposto por [Karner, 1993], foi modificado de 20 (sugerido por Karner, criador do método, para quem não tem dados históricos) para 18.

iii. Instrumentação. A ferramenta CASE utilizada foi substituída em razão das limitações encontradas.

2.4 Projeto Utilizado na Fábrica de Software

O projeto utilizado na fábrica foi o desenvolvimento do sistema SI Alocação Plus. Esse sistema visa solucionar o problema de alocação de salas (aulas, laboratórios, auditórios), professores, alunos e disciplinas, através da eliminação de conflitos, utilizando técnicas de alocação de recursos, como, por exemplo, *Job-shop problem* [Jain, 1998] e *Time Scheduling* [Schaerf, 1995]. A Figura 3 mostra a visão geral do sistema.

O IEC, Informatics Educacional Center, é uma instituição privada que possui unidades de ensino em diversos países. A partir de setembro, esta instituição estará implantando novas unidades no Estado de Pernambuco, que irão necessitar de soluções de gestão acadêmico-financeira.

O instituto necessita de um sistema que permita realizar as seguintes funcionalidades:

- Cadastro, alteração, exclusão e consulta de seus professores;
- Cadastro, alteração, exclusão e consulta de suas disciplinas;
- Cadastro, alteração, exclusão e consulta de suas salas;
- Cadastro, alteração, exclusão e consulta das alocações;
- Sugestão de alocações para os clientes.

Figura 3. Descrição do Projeto de Desenvolvimento.

2.5 Instanciação da Fábrica de Software

Seleção dos Participantes: para a execução do projeto, os participantes foram alocados nos perfis apresentados na seção 2.2. Entretanto, alguns participantes desempenharam mais

de um papel, a fim de evitar a ociosidade e melhorar o desempenho geral da fábrica. Os participantes selecionados foram alunos dos cursos de Mestrado e Doutorado em Ciência da Computação da Universidade Federal de Pernambuco, realizando a disciplina engenharia de software. Os participantes foram selecionados por conveniência, de acordo com seus conhecimentos e seu perfil, representando um subconjunto não aleatório do universo de alunos. A Figura 4 mostra a estrutura dos participantes da fábrica de software.

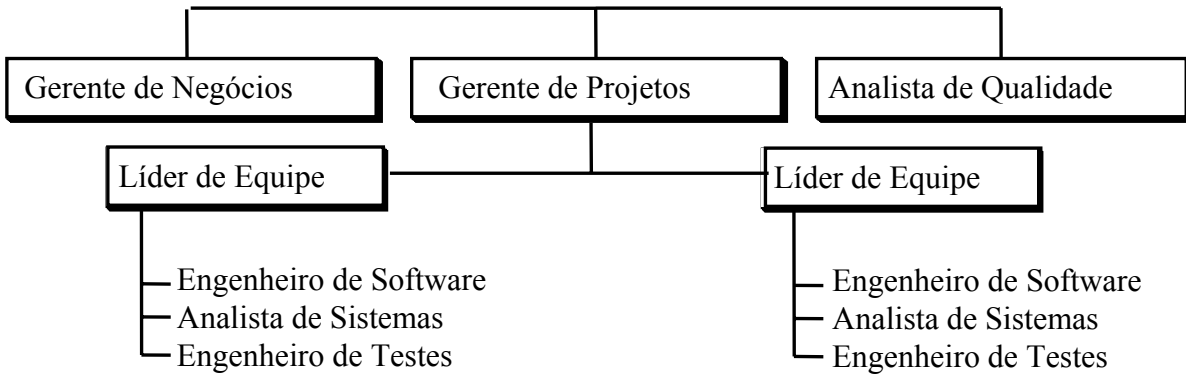


Figura 4. Estrutura dos participantes da Fábrica.

Capacidade aleatória: a seleção dos participantes do projeto não foi aleatória, pois considerou o conhecimento e o perfil dos indivíduos. De posse das características dos participantes, o processo de alocação foi realizado, entretanto, de comum acordo com todos os membros.

2.6 Execução da Fábrica de Software

Realização: o projeto foi realizado durante parte de um curso de pós-graduação em Ciência da Computação, durante o primeiro semestre do ano letivo de 2003, na Universidade Federal de Pernambuco. O prazo estipulado para execução do projeto foi de aproximadamente dois meses, após a realização do piloto.

Treinamento: os participantes que realizaram o projeto foram treinados antes da realização do projeto. Os treinamentos foram divididos por fases do processo de desenvolvimento, a saber: desenvolvimento de componentes (abordagem, ferramental necessário) e testes e validação.

Participantes: os participantes incluíram os alunos do mestrado e doutorado do programa de Pós-Graduação em Ciência da Computação. Dos 11 participantes, constavam 9 alunos de mestrado e 2 alunos de doutorado. Dez participantes possuíam experiência de desenvolvimento, ou em ambiente acadêmico ou industrial, enquanto um participante possuía pouca experiência. A Tabela 1 apresenta um resumo da formação e da experiência dos participantes da fábrica.

Tabela 1. Perfil dos participantes da Fábrica

ID	Formação	Experiência	Perfis
1	Doutorando	Indústria	Análise, Testes
2	Doutorando	Indústria	Implementação, Suporte
3	Mestrando	Indústria	Testes, comercial
4	Mestrando	Indústria	Implementação, Testes
5	Mestrando	Academia	
6	Mestrando	Indústria	Qualidade, Testes
7	Mestrando	Academia	
8	Mestrando	Academia	Gerência, Qualidade
9	Mestrando	Indústria	Comercial, Análise, Implementação
10	Mestrando	Indústria	Análise, Implementação
11	Mestrando	Indústria	Implementação, Testes

Custo do Projeto: como os participantes da fábrica foram alunos da própria universidade e os equipamentos utilizados foram os dos laboratórios, o custo de realização do projeto se concentrou no custo de planejamento.

Custo de Planejamento: o planejamento é, certamente, a etapa mais custosa para a definição de uma fábrica de software. O planejamento da fábrica se estendeu durante quatro meses, entre abril e julho do ano de 2003, concentrando-se com a cultura de fábricas de software, definições de processos, artefatos, até a execução do projeto piloto.

2.7 Análise dos Resultados

Avaliação do Treinamento: o treinamento foi aplicado apenas para as equipes de implementação e testes. O treinamento ocupou duas horas de duração, utilizando um conjunto de slides e, ao final, foi distribuído um material para acompanhamento do processo de desenvolvimento dos componentes. Os participantes da equipe de testes sugeriram a necessidade de mais treinamento para os próximos projetos, em virtude da falta de conhecimento para realização desta atividade.

Avaliação Quantitativa: a execução de todo o projeto resultou num total de quinhentos e quarenta e quatro (544) horas, distribuídas do seguinte modo:

- *Comercial: 70 horas (12,87% aprox.)*
- *Planejamento e Gerenciamento: 69 horas (12,68 % aprox.)*
- *Desenvolvimento de Componentes: 352 horas (64,71% aprox.)*
- *Testes: 53 horas (9,74% aprox.)*

A fase de Desenvolvimento de Componentes foi a mais dispendiosa. Durante esta fase, alguns participantes, que já tinham executados seus perfis, foram realocados, a fim de aumentar a produtividade do desenvolvimento.

Utilidade do Processo: todos os participantes ressaltaram a utilidade do processo utilizado pela fábrica. Entretanto, algumas sugestões de melhoria foram retificadas, principalmente, em algumas etapas do processo de desenvolvimento de componentes, a fim de possibilitar a utilização de outras ferramentas CASE, assim como permitir apenas o desenvolvimento de componentes de negócio, considerando os requisitos funcionais, como, por exemplo, distribuição, opcional.

2.8 Lições Aprendidas

Nos projetos subseqüentes, alguns pontos devem ser considerados, vistos como limitações de sua primeira execução. Esta seção apresenta as lições aprendidas através de três grandes marcos da fábrica de software.

2.8.1 Definição da Fábrica

Melhoria do Processo: Devido ao envolvimento dos participantes em outras atividades (dedicação parcial), a gerência da fábrica de software precisou ajustar algumas tarefas, de modo que permitisse a sua execução de maneira distribuída, utilizando-se de ferramentas apropriadas. Todavia, em alguns instantes, a presença dos participantes tornava-se extremamente necessário, porém, devido à dedicação parcial, muita destas atividades eram realizadas em outras datas, comprometendo, em alguns pontos, o cronograma definido.

Outro ponto que causou extrema dificuldade durante a etapa de definição da fábrica foi à carência de conhecimento sobre fábricas de software. Grande parte dos participantes não tinha experiência sobre o tema, ocasionando uma fase de pré-definição através de pesquisas na literatura e reuniões internas, a fim de amadurecer o conceito. Entretanto, a própria literatura era carente de relatos de experiências mostrando a criação de fábricas de software, muitas vezes, até por questões estratégicas.

Adicionalmente, dois outros aspectos mereceram considerações especiais. O primeiro foi a definição do escopo da fábrica. Por não termos conhecimento prévio da dimensão do sistema que seria proposto, o processo da fábrica teria que permitir a instanciação tanto para projetos de pequeno porte, quanto de grande porte. Isso influenciou em diversos aspectos ferramentais e de processo a ser utilizado, que só com a demanda definida foi possível delinear. Por fim, devido à incompatibilidade de horário entre os participantes e clientes, foi difícil estabelecer horários para reuniões, levando a dificuldades para o melhor esclarecimento das dúvidas que surgiram.

2.8.2 Execução e ajustes do Projeto Piloto

A execução do piloto foi de extrema utilidade, pois a obtenção de resultados proporcionou a possibilidade de avaliar as tarefas do primeiro mês de atividade da fábrica e identificar os possíveis problemas e dificuldades encontradas, buscando ajustá-las à realidade e ao contexto da empresa. Contudo, alguns problemas de planejamento ocorreram.

Planejamento: Devido a alguns recessos, algumas atividades de desenvolvimento tiveram que ocorrer em paralelo (especificação, projeto e implementação dos componentes), ocasionando uma certa dificuldade em relação à etapa subseqüente, executar testes. Além disso, devido a uma demanda na última semana do projeto piloto, ocorreu uma sobrecarga de alguns participantes da equipe de desenvolvimento. Por fim, uma vez que a fábrica não possuía qualquer dado histórico de projetos anteriores, a definição de prazos e perfis do projeto piloto ficou um pouco comprometida, porém, não causou atrasos no cronograma definido. Assim, para os projetos subseqüentes espera-se que boa parte dessas limitações possa ser contornada, pois a fábrica já possui dados históricos, que mesmo iniciais, serão de grande utilidade.

2.8.3 Execução do Projeto

Ao termino do projeto piloto, algumas ações foram planejadas a fim de contornar as limitações encontradas. Assim, os seguintes aspectos foram definidos: a realocação de alguns perfis, a presença efetiva dos líderes das equipes (analistas, engenheiros e testes) e,

por fim, a uniformidade de horas gasta pelos participantes, evitando a sobrecarga de trabalho, principalmente, no setor de implementação.

Planejamento: Durante a execução do projeto, um grande problema que surgiu foi o paralelismo com outras atividades acadêmicas. Como o projeto foi realizado próximo ao fim do semestre, os participantes tiveram que paralelizar ao máximo suas tarefas na fábrica, com as demais atividades da universidade, causando, em alguns momentos, a realocação de alguns participantes. Finalmente, devido a um certo conflito para entendimento da natureza do problema e das reais necessidades do cliente, ocorreu um maior dispêndio de tempo na fase comercial do processo de desenvolvimento.

3. Trabalhos Relacionados

Alguns trabalhos envolvendo fábricas de software existem na literatura. Nesta seção, apresentaremos alguns deles, enfatizando suas diferenças e similaridades em relação ao presente trabalho.

Em [Siy, 2001], os autores apresentam uma vasta coleção de dados de entrevistas, questionários e métricas de software, a fim de descrever como criar e manter uma organização efetiva, centrada em processos de desenvolvimento de software.

Fernström [1992] descreve o projeto *Eureka Software Factory* (ESF), que tem por objetivo permitir a criação de um mercado para produtos CASE, que possam ser configurados para aplicações específicas, ajudando os construtores de fábricas a integrar os produtos e informações utilizadas ao longo do processo.

A ESF representa um esforço conjunto de treze entidades parceiras de cinco países Europeus, entre as quais encontram-se instituições de pesquisa, fabricantes de computadores, desenvolvedores de ferramentas CASE e sistemas.

A grande diferença deste trabalho em relação aos demais, está no relato das experiências com uma fábrica de software, desde a sua fase de definição até as lições aprendidas durante todo o processo.

4. Conclusão e Trabalhos Futuros

Neste artigo, apresentamos o processo de criação de uma fábrica de software discutindo, detalhadamente, desde as etapas de definição e planejamento, até as lições aprendidas incorporadas durante o processo. Embora o trabalho seja mais um relato envolvendo fábricas de software, a sua maior contribuição está numa detalhada descrição de como criar uma fábrica de software sem muitos recursos físicos e financeiros, aspecto este, não mencionado com muita frequência pela literatura.

Como trabalhos futuros, pretende-se realizar projetos de grande porte, a fim de realizar alguns ajustes e refinar os processos anteriormente definidos.

Referências Bibliográficas

[Aaen, 1997] Aaen, I., Bottcher, P., Mathiassen, L., *The Software Factory: Contributions and Illusions*, In the Proceedings of the Twentieth Information Systems Research Seminar, Scandinavia, Oslo, 1997.

[Almeida, 2003] Almeida, E. S., Bianchini, C. P., Prado, A. F., Trevelin, L. C., *IPM: An Incremental Process Model for Distributed Component-Based Software Development*, In

The 5th International Conference On Enterprise Information Systems (ICEIS), Angers - France. ACM Press, 2003.

[Cleland, 2000] Cleland, D., *A Guide to the Project Management Body of Knowledge*, 2000.

[Fernström, 1992] Fernström, C., Närfelt, K., Ohlsson, L., *Software Factory Principles, Architecture, and Experiments*, IEEE Software, 1992

[Frakes, 1995] Frakes, W. B., Fox, C. J., *Sixteen Questions about Software Reuse*, In the Communications of the ACM, Vol. 38, June 1995.

[Jacobson, 2001] Jacobson, I., et. al., *The Unified Software Development Process*, Addison-Wesley, 2001.

[Jain, 1998] Jain, A., Meeran, S., *A State-of-the-art Review of Job-shop Scheduling Techniques*, Technical report, University of Dundee, Dundee, 1998.

[Karner, 1993] Karner, G., *Resource Estimation for Objectory Projects*, Copyright Rational Software, 1993.

[Schaerf, 1995] Schaerf, A., *A survey of automated timetabling*, Report, CWI, Amsterdam, 1995.

[Siy, 2001] Siy, H. P., Mockus, A., Herbsleb, J. D., Krishnan, M., Tucker, G. T., *Making The Software Factory Work: Lessons from a decade of experience*, In the Seventh International Symposium on Software Metrics, London, England, April 2001.