

Deploying on the Grid with DeployWare

<http://fdf.gforge.inria.fr/>

Original: Areski Flissi, Jérémy Dubus, Nicolas Dolet and
Philippe Merle

Apresentador: Amadeu Andrade Barbosa Júnior
ajunior@inf.puc-rio.br
INF2556 – Seminários de Sistemas Distribuídos
Departamento de Informática
PUC-Rio

03 de setembro de 2008

Agenda

- 1 Motivação
- 2 DeployWare
 - Internamente
 - Contribuições
- 3 Considerações Finais

Implantação em Grades - tarefas

Administradores de sistemas normalmente adotam soluções próprias para gerir:

- acesso remoto (ssh, telnet, jssh)
- instalação e desinstalação dos middlewares (corba, ccm, j2ee, globus)
- bibliotecas dependentes e utilitários de suporte (apt-get, rpm, .tgz, installshield)
- fluxo de execução e parada dos programas (scripts para start/stop e monitoramento)
- transferências de dados de simulação (sistemas de arquivos compartilhados ou scripts de coleta)

Quase toda a gestão da implantação precisa ser alterada caso use-se uma outra tecnologia

Implantação em Grades - desafios

complexidade : cenários com milhares de nós não pode ser gerido por centenas de pessoas, as tarefas automáticas precisam ser **orquestradas**

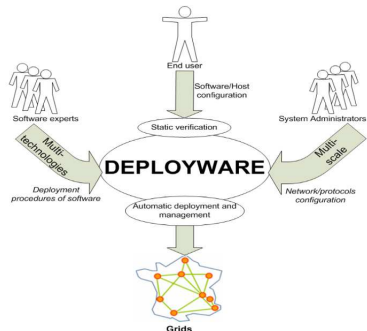
heterogeneidade : não há apenas máquinas diferentes, mas sistemas operacionais, protocolos de redes e softwares diferentes

validação : validar previamente se a implantação em larga escala funciona é fundamental para o processo de desenvolvimento

escalabilidade : os recursos físicos de uma máquina gestora da implantação podem ser facilmente esgotados, é preciso escalar também o processo de implantação

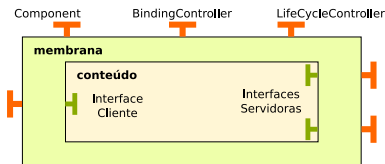
DeployWare [Flissi et al., 2008]

- Dispõe de uma DSML (*Domain Specific Modeling Language*) que dita um **meta-modelo de implantação**
- Para cada tecnologia **cria-se um modelo concreto** representado em uma ADL (*Architecture Description Language*)
- Dispõe de uma máquina virtual que interpreta o roteiro de implantação da ADL
- Todos elementos do modelo e a implementação da máquina virtual são **componentes Fractal!**



DeployWare: Componentes Fractal

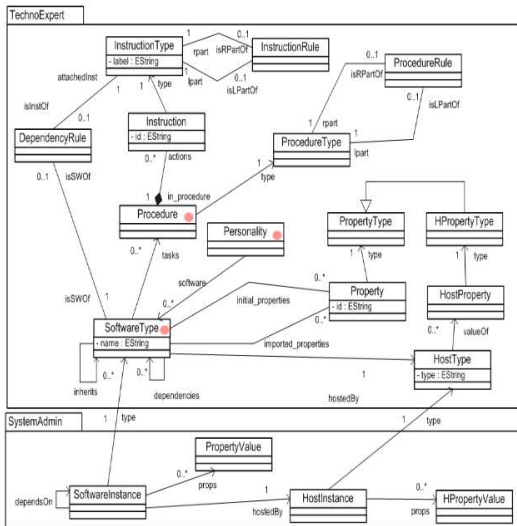
- Fractal define uma *membrana* com interfaces para inspecionar e reconfigurar seus elementos internos
- Dentro da membrana está o *conteúdo* = conjunto de sub-componentes
- Membrana é composta por *controladores* que gerenciam as ligações com sub-componentes
- Controladores podem agir como interceptadores sobre as operações solicitadas e ditar os *níveis de controle*



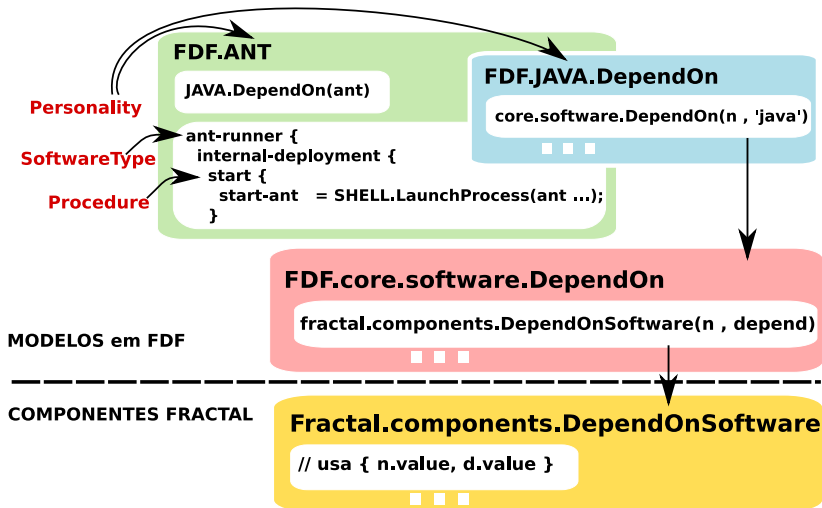
DeployWare: Meta-modelo (1)

- **Personality** : define a tecnologia
- **SoftwareType** : define um artefato de implantação que possui **Properties**
- **Procedure** : define um conjunto de **Instructions** a serem executadas
- **Dependencies** : representa a dependência entre dois softwares
- Dois pacotes principais:
 - **TechnoExpert**, voltado a quem porta uma nova tecnologia
 - **SystemAdmin**, voltado aos administradores e usuários finais que descrevem seu ambiente e instâncias de software

DeployWare: Meta-modelo (2)



DeployWare: Relação entre Modelo, Meta-modelo e Componentes



DeployWare: Exemplo de Sistema de Banco

```

BankExample {
  Hosts = INTERNET.NETWORK { bank = INTERNET.HOST {...} }
  Software {
    bank {
      ear = JOnAS.EAR(/filebank.ear) { jonas = /Software/bank/jo
      jonas = JOnAS.SERVER {
        archive = JOnAS.ARCHIVE(/filejonas.tgz);
        root = JOnAS.ROOT(/usr/local/jonas);
        host = Hosts/bank;
        properties { http-port = HTTP.PORT(9000); }
        dependencies { Software/bank/mysql; }
      }
      mysql = MySQL.SERVER {
        archive = MySQL.ARCHIVE(/filemysql.tgz);
        home = MySQL.HOME(/usr/local/MySQL);
        host = Hosts/bank;
      }
    }
  }
}

```

Interface de Monitoramento

The screenshot displays the DeployWare Explorer interface. On the left, a tree view shows the deployment structure:

- Fractal RMI Registries
 - CCGRID08
 - compute
 - Hosts
 - main_node
 - g5k-nodes
 - Software
 - Main-Services
 - JRE-Software
 - ORB-Software
 - Middleware
 - OpenCCM_Servers_Deployment
 - NameService
 - OpenCCM_DCI
 - Assembly_Factory
 - OpenCCM-Node-Managers
 - SERVER-1
 - openccm
 - dcl
 - host
 - ior_ns
 - ior_dcl
 - ior_proxy
 - nm
 - dependencies
 - SERVER-2
 - SERVER-3
 - SERVER-4
 - SERVER-5
 - SERVER-6

On the right, a dependency graph shows the relationships between components. A central box for 'SERVER' lists actions: Install (Ctrl-I), Start (Ctrl-S), Stop (Ctrl-S), and Uninstall (Ctrl-U). Arrows labeled 'depend' and 'host' connect this box to various components like 'dcl', 'ns', 'comanche', 'openccm', 'java', and 'chuque-6.lille.grid5000.fr'. The graph also shows dependencies between 'dcl', 'ns', 'comanche', and 'openccm'.

At the bottom, the status bar indicates: Explorer - loading C:\papers\ccgrid08\CCGRID08.fdf... Explorer - C:\papers\ccgrid08\CCGRID08.fdf loaded. Deployment status is INSTALLED.

Escalabilidade

- A partir do console do DeployWare é possível implantar mais servidores do próprio DeployWare, devido à existência de um **Personality** do próprio framework
- Verificaram ganhos no tempo total da implantação usando a versão distribuída do DeployWare já que na versão de único servidor, a quantidade de sockets e carga prejudicava a implantação dos milhares de nós

Trabalhos relacionados

ProActive, GoDIET Só permitem implantação dos seus próprios middlewares.

Jade Depende da escrita de envólucros para implantação, permite implantar códigos de qualquer linguagem e assume serviços pré-existentes como registry. **No DeployWare as descrições são mais simples.**

ADAGE Há separação entre descrições dos recursos (hardware) e dos softwares a implantar, mapeamento feito no planejador, possui apenas três fases: instalação, lançamento e configuração. **O meta-modelo do DeployWare é mais flexível.**

OMG D&C Especificação de um fluxo de implantação que suporta diferentes modelos de componentes. **Não trata a orquestração, validação e escalabilidade da implantação do middleware.**

Discussão final

- + A ADL possui facilidades para instanciação (forEach,ifThenElse);
- + A ADL simplifica o processo de descrição das aplicações e facilita o mapeamento nos componentes Fractal;
- + A validação estática do deploy em etapa anterior à execução. Permite à equipe ter um ambientes de testes mais próximo do real antes de liberar os *releases*;
- ? Quanto o DeployWare (por ser java) consome de memória/cpu/recursos? O SCS é *lightweight* e já dispõe de *serviços importantes: Suspensão, Intercepção, Autenticação e Sessão, úteis a tarefas como re-deployment, co-deployment, healing...*
- ? O DeployWare não tem uma ligação direta com sistemas de empacotamento. Permite instalação via dpkg, rpm, untar, unzip. *Obriga desenvolvedor/administrador garantir que dependências externas estejam resolvidas.*



Flissi, A., Dubus, J., Dolet, N., and Merle, P. (2008).

Deploying on the grid with DeployWare.

In *Proceedings of the 8th International Symposium on Cluster Computing and the Grid (CCGRID'08)*, pages 177–184. IEEE Computer Society.