Uma Infraestrutura Para a Execução de Workflows WS-BPEL em Clusters de Clusters

Thiago Alvarenga Lechuga

21/10/2010

Orientadora: Maria Beatriz Felgar de Toledo
Agência Financiadora: FAPESP

Sumário

Introdução

Introdução

- Motivação para os *Clusters* de *Clusters*.
- Tarefas complexas precisam ser facilmente compostas e executadas, mantendo suas dependências.
  - *Workflow*?
  - Padrões atuais para a composição de serviços em sistemas distribuídos heterogêneos ainda são inadequadas.
  - Requisitos específicos que devem ser considerados.
  - BPEL é um forte candidato (Serviços Web).

Objetivos

- Extensão à linguagem WS-BPEL.
- Especificação e seleção de recursos baseada em:
  - Requisitos de QoS;
  - Recursos computacionais requeridos para a execução de cada serviço.
- Proposta de arquitetura para a execução de *workflows* em CoCs.
- Implementação da arquitetura proposta em um ambiente de produção.
- Especificação e implementação de um problema real no ambiente criado.

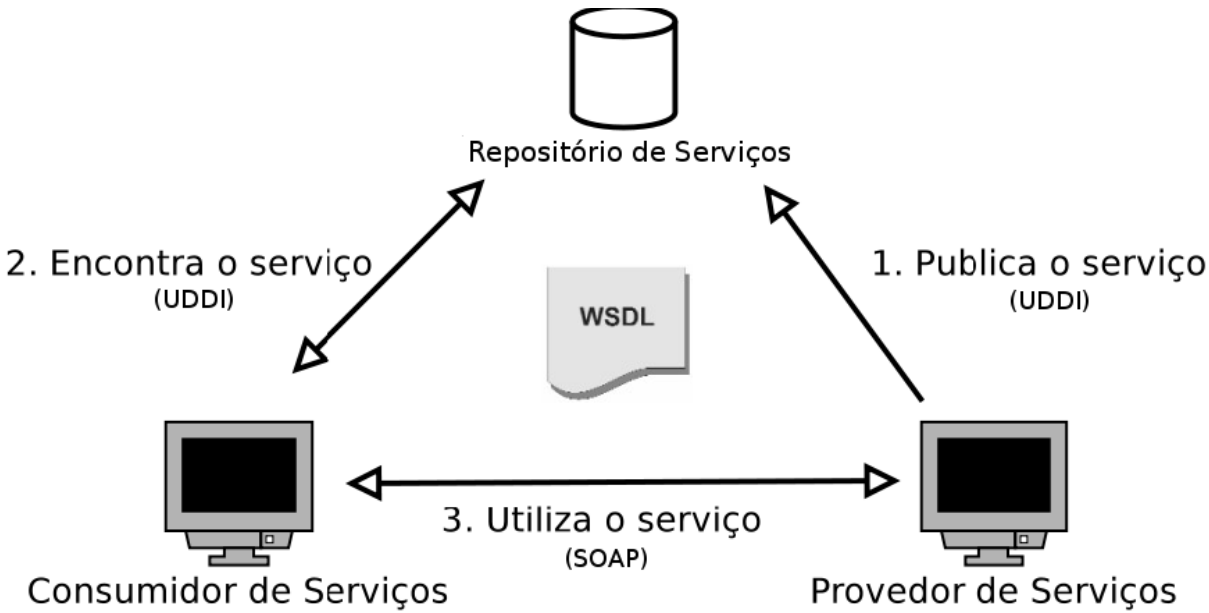
Fundamentos

SOA e Serviços Web

- Emergiram como uma solução para a comunicação entre aplicações.
- SOA faz uso extensivo da tecnologia de WS de maneira a definir uma estrutura distribuída, independente de plataforma e linguagem de programação.
- Interoperabilidade.

Camadas Básicas

- Descoberta: UDDI.
- Descrição: WSDL.
- Comunicação: SOAP.



Padrões SOA estabelecidos.

WS-Addressing

- Especificação para que serviços Web comuniquem informações de endereçamento.
- Maneira padronizada de incluir informações de roteamento da mensagem no cabeçalho de mensagens SOAP.
- Estrutura:
  - MessageID;
  - ReplyTo;
  - RelatesTo.
- Comumente utilizado para a comunicação de WS em modo assíncrono.

WS-Addressing - Requisição

```
1  <soapenv:Envelope>
2
3      <soapenv:Header>
4          <add:MessageID>ID-Unico</add:MessageID>
5          <add:ReplyTo>
6              <add:Address>http://127.0.0.1/</add:Address>
7          </add:ReplyTo>
8      </soapenv:Header>
9
10     <soapenv:Body>
11         <ewe:executar>
12             <comando>parametro1</comando>
13             <arqOutput>parametro2</arqOutput>
14         </ewe:execSharcnet>
15     </soapenv:Body>
16
17 </soapenv:Envelope>
```

Exemplo de uma mensagem SOAP utilizando WS-Addressing.

WS-Addressing - Resposta

```
1 <soapenv:Header>
2     <add:RelatesTo>ID-Unico</add:RelatesTo>
3 </soapenv:Header>
```

Trecho do cabeçalho de uma requisição SOAP de resposta utilizando WS-Addressing.

WS-Policy

- Especificação de políticas para serviços Web.
- Flexível e extensível.
- Especificam características importantes para a seleção e utilização de serviços Web (ex: características não-funcionais).

WS-Policy - Exemplo

```
1  <wsp:Policy>
2      <wsp:ExactlyOne>
3          <wsp:All>
4              <qosp:ResponseTime
5                  xmlns:qosp="../schema/qospolicy"
6                  operation="get"
7                  specification="uddi:qos:attribute:
8                  responsetime">45</qosp:ResponseTime>
9          </wsp:All>
10     </wsp:ExactlyOne>
11 </wsp:Policy>
```

Exemplo de uma política WS-Policy.

WS-BPEL

- Padrão de fato para a composição de serviços Web.
- Linguagem baseada em XML que descreve um *workflow* de serviços.
- Descreve o relacionamento entre os diversos serviços Web participantes da composição.

WS-BPEL - Estrutura Principal

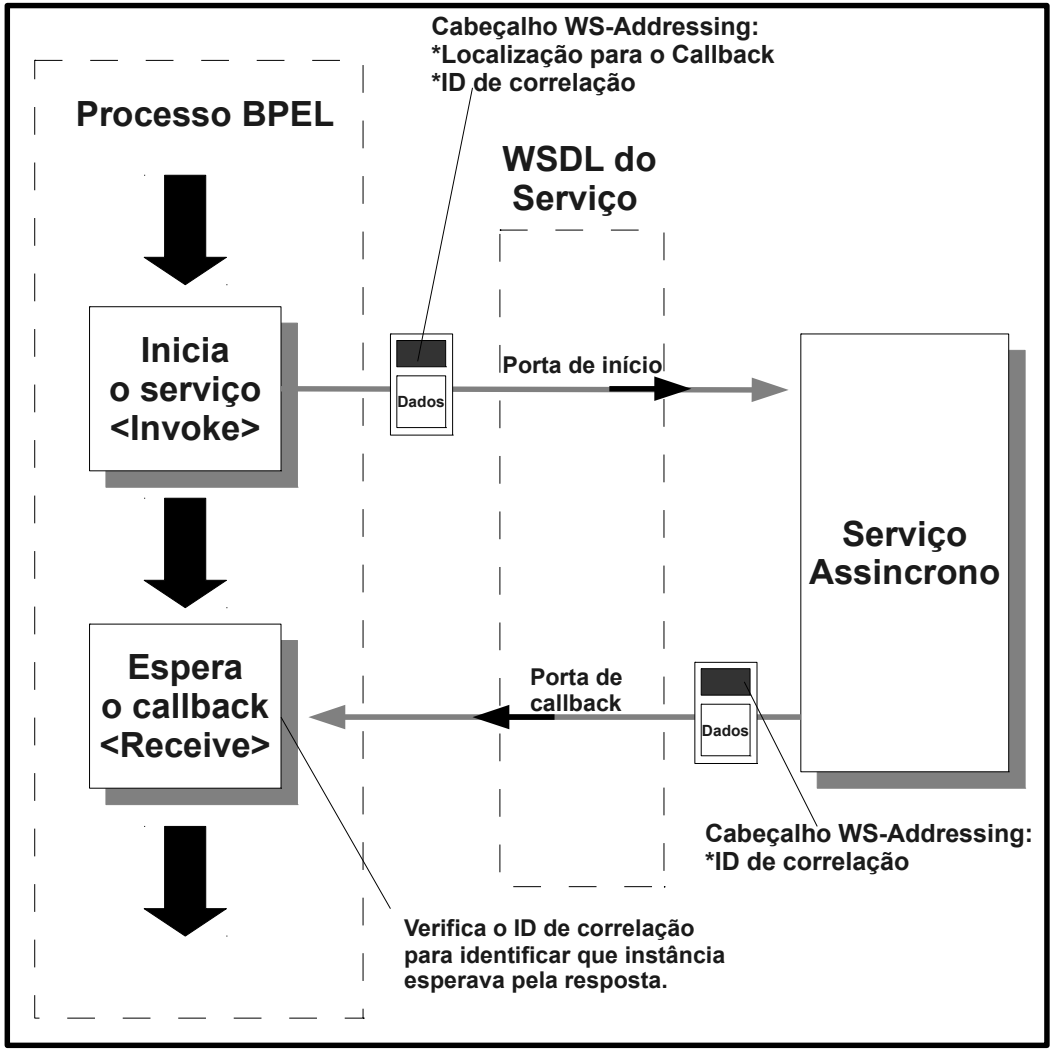
- **PartnerLinks**: Define os parceiros que interagem com o processo de negócio durante a execução. Identificar funcionalidades que devem ser oferecidas por cada serviço.
- **Variables**: Define as variáveis de dados usadas pelo processo.
- **Activities**: Descrição do comportamento normal para a execução do processo de negócio.
  - **Basic Activity**: Tipo de atividade usado para executar alguma operação (`invoke`, `receive` e `reply`).
  - **Structured Activity**: Tipo de atividade usado para agrupar atividades básicas dentro de estruturas de fluxo (`while`, `pick`, `flow`, `sequence`, `switch` e `scope`).

WS-BPEL - Exemplo

```
1  <process name="Sharcnet">
2      <partnerLinks>
3          <partnerLink name="PLSharcnet" partnerLinkType="AsyncSharcnet"
               partnerRole="R1" />
4      </partnerLinks>
5
6      <variables>
7          <variable name="RunOut" messageType="runResponse" />
8          <variable name="RunIn" messageType="run" />
9      </variables>
10
11     <sequence>
12         <receive name="ReceiveInicio" partnerLink="PLinkClient" operation="
               operationA" portType="MyPTClient" variable="OperationAIn" />
13
14         <assign name="Assign0">
15             <copy>
16                 <from>'VALOR1'</from>
17                 <to>$RunIn.parameters/arg0</to>
18             </copy>
19         </assign>
20
21         <invoke name="InvokeSharcnet" partnerLink="PLSharcnet" operation="
               execSharcnet" inputVariable="ExecSharcnetIn" />
22     </sequence>
23 </process>
```

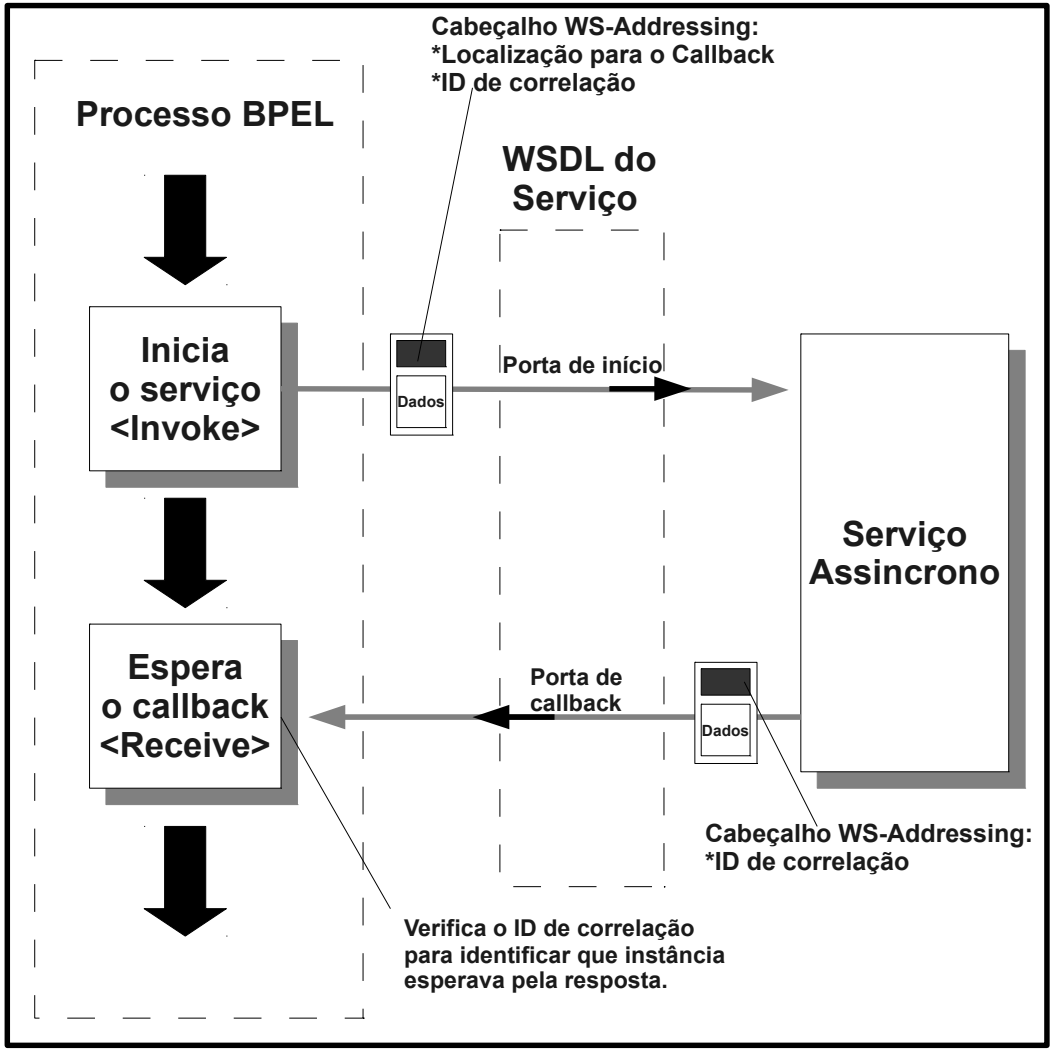
Exemplo de um *workflow* descrito na linguagem BPEL.

Execução de serviços assíncronos - BPEL



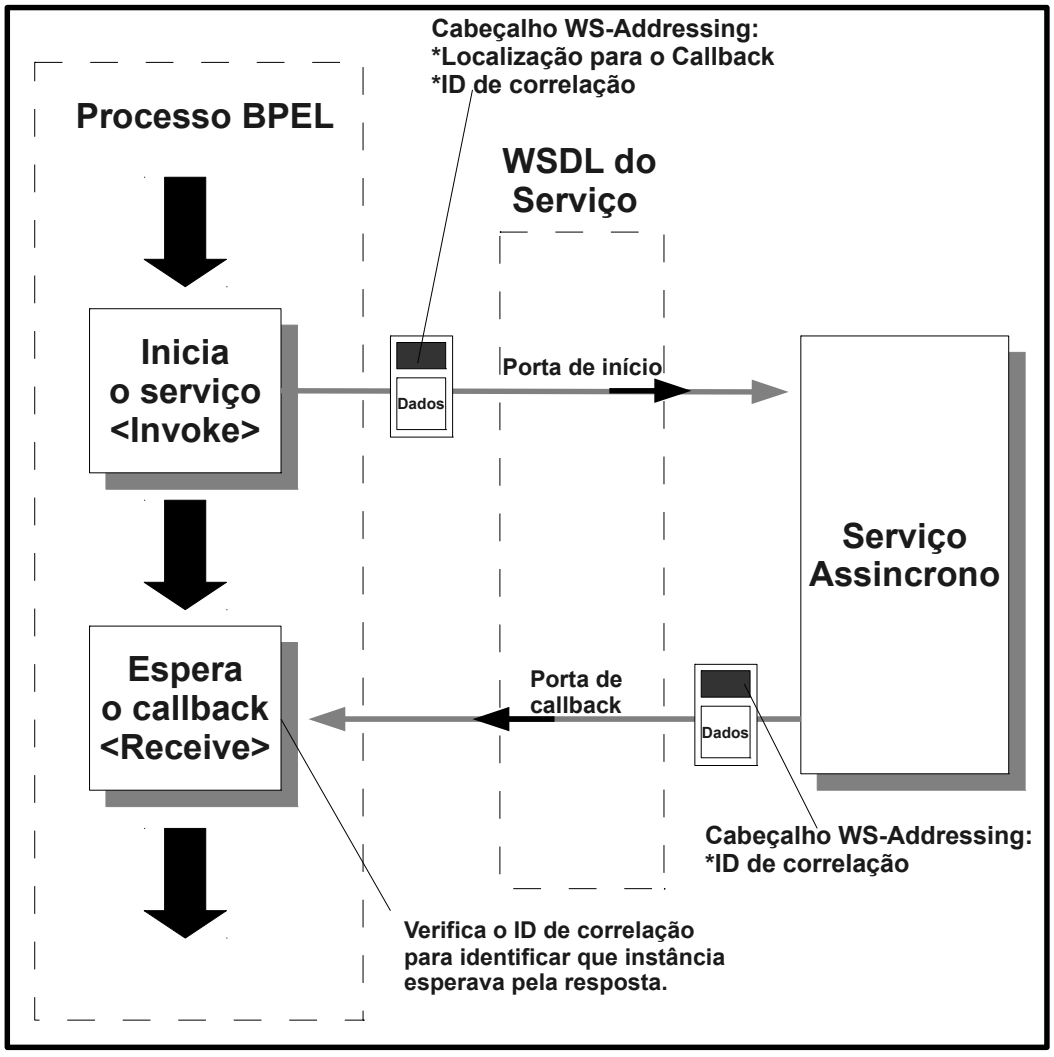
1 - No BPEL, uma atividade `invoke` inicia um serviço (SOAP);

Execução de serviços assíncronos - BPEL



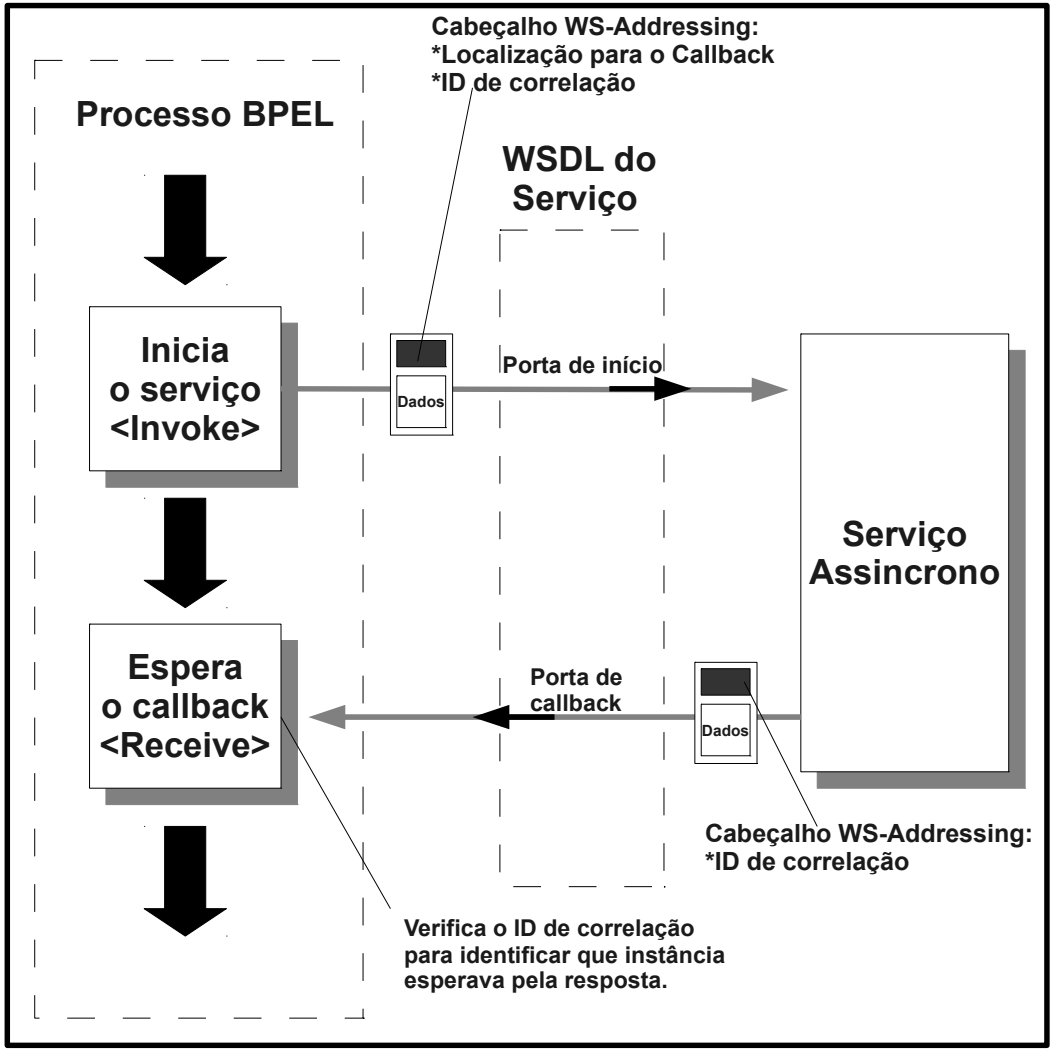
2 - Cabeçalho WS-addressing é adicionado, informando o endereço do serviço para *callback*, e o identificador de correlação;

Execução de serviços assíncronos - BPEL



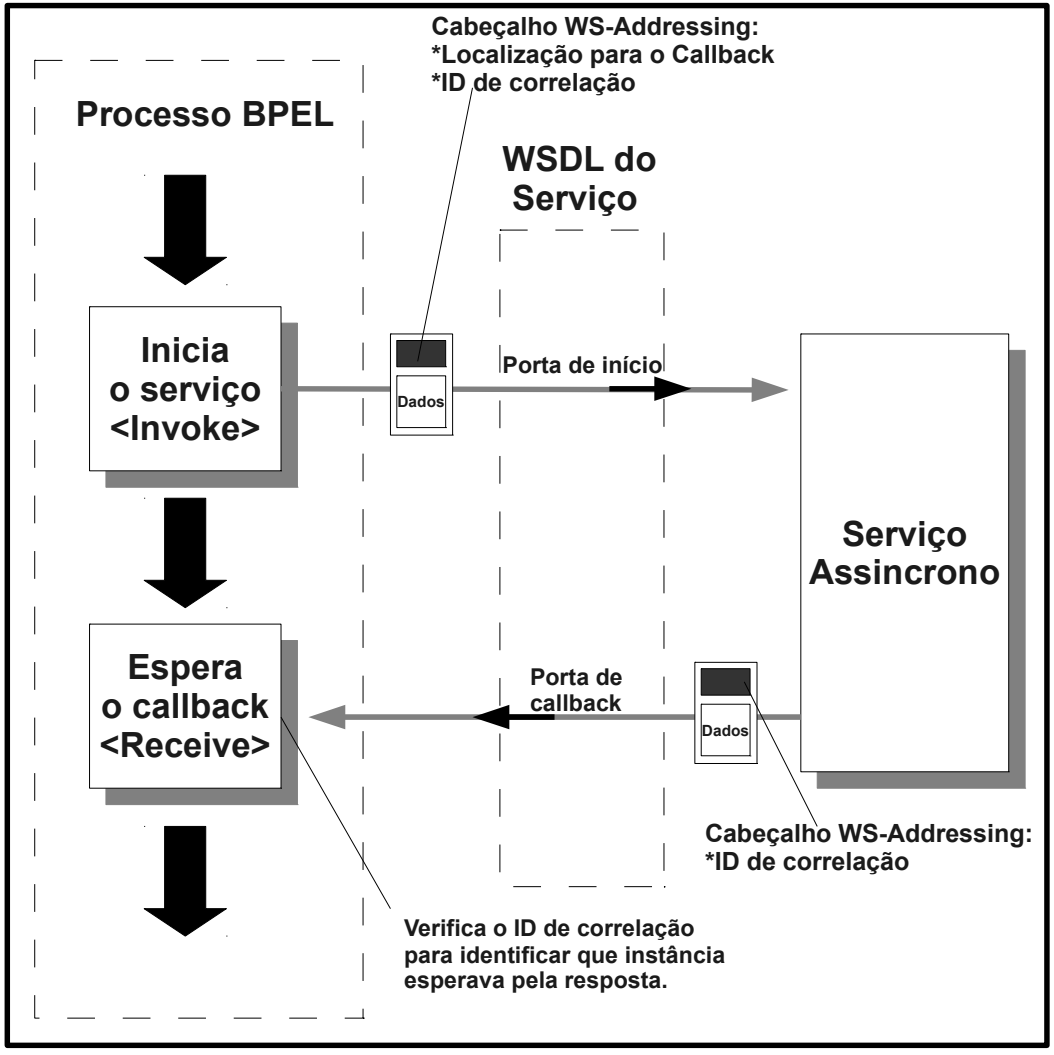
3 - Usando a descrição WSDL do serviço, uma porta é iniciada e os dados para o serviço assíncrono são enviados;

Execução de serviços assíncronos - BPEL



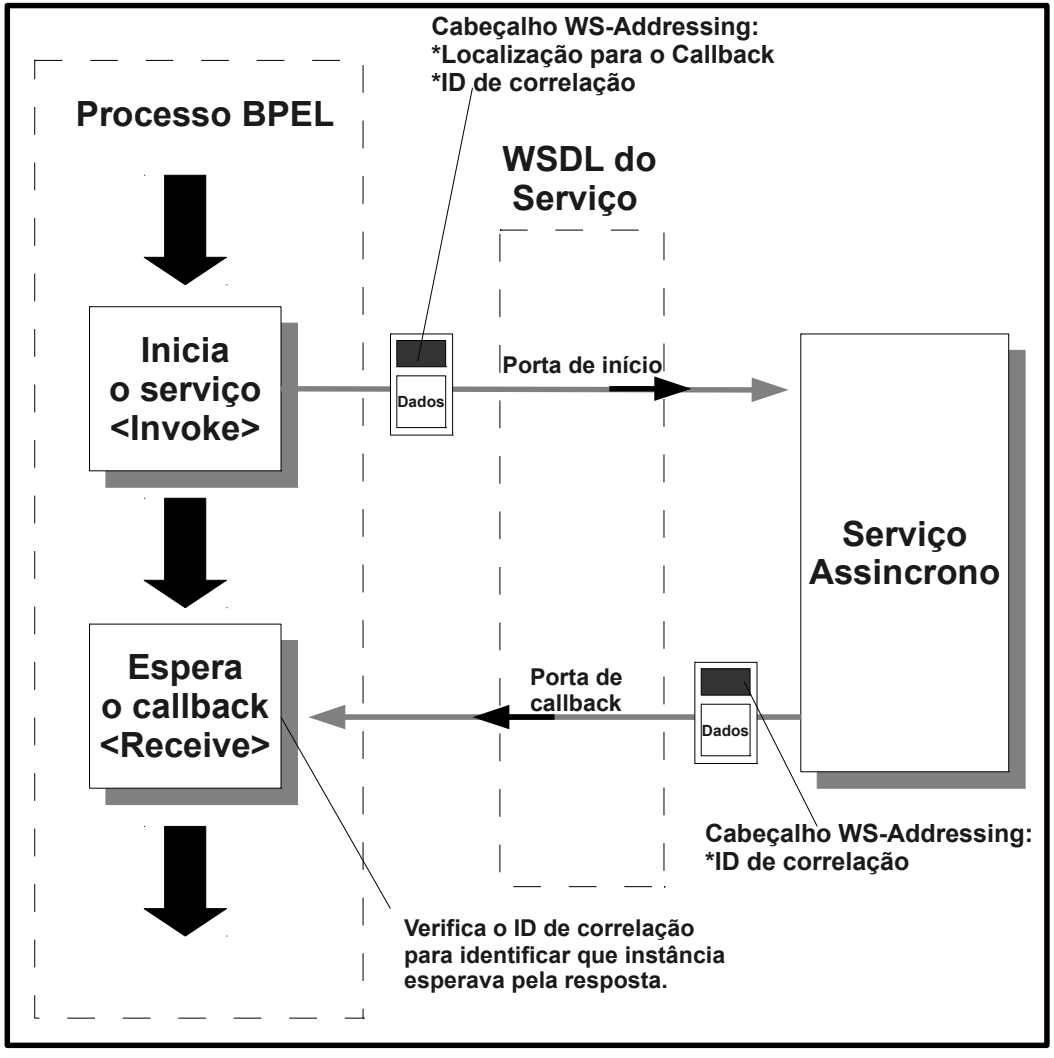
4 - O serviço assíncrono processa a requisição e envia uma resposta de volta (porta de *callback*).

Execução de serviços assíncronos - BPEL



5 - O cabeçalho WS-addressing é montado com o identificador de correlação apropriado (o mesmo da mensagem `invoke`);

Execução de serviços assíncronos - BPEL



6 - A máquina de composição BPEL verifica o identificador de correlação e identifica a instância que aguardava.

Execução de serviços assíncronos - BPEL

Tempo de vida da chamada/resposta SOAP desacoplado do tempo de vida do protocolo HTTP, permitindo a execução de processos muito demorados.

Trabalhos Relacionados

Execução de WFs em Sistemas Distribuídos Heterogêneos

Autor	Arquitetura Alvo			Outras Propriedades		
	CoC	WSRF	OGSI	BPEL	Descoberta	Políticas
Leymann	Não	Não	Não	Sim	Não	Não
Slomiski	Não	Sim	Sim	Sim	Não	Não
Emmerich	Não	Não	Sim	Sim	Não	Não
Ezenwoye	Não	Sim	Não	Sim	Sim	Não
Ma	Não	Sim	Não	Sim	Não	Não
Lechuga	**Sim**	**Não**	**Não**	**Sim**	**Sim**	**Sim**

Escalonamento Global em CoCs

Autor	Problemas		Vantagens	
	Central.	Alterações	Implement.	Compatib.
Chau	Não	Sim	Não	Não
Takpé	Sim	Não	Não	Não
Qin	Sim	Sim	Não	Não
Hunold	Não	Não	Não	Não
Beltrán	Não	Sim	Sim	Não
Lechuga	**Não**	**Não**	**Sim**	**Sim**

Infraestrutura para a Execução de WFs em CoCs

Estendendo o WS-BPEL para especificação de QoS

- Seleção de serviços baseada em especificações de QoS e Recursos.
  - Inclusão de informações de recursos requeridos para a execução.
  - Requisitos dos usuários e capacidade dos provedores.
- Informações de QoS especificadas como políticas em WS-Policy e armazenadas em um repositório.
- Funcionalidades e atributos de QoS em arquivos separados (WS-PolicyAttachment).

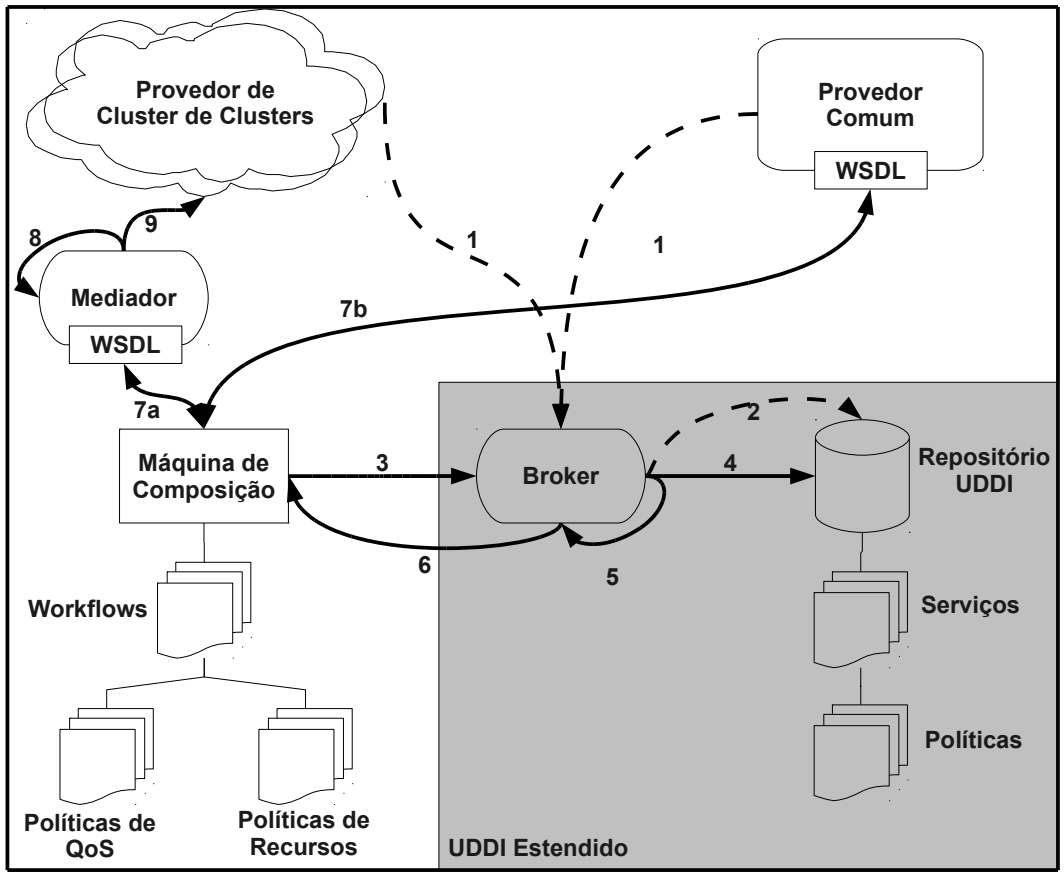
Estendendo o WS-BPEL para especificação de QoS

```
1 <b:invoke
2     partnerLink="Seller"
3     portType="a:Purchasing"
4     operation="Purchase"
5     inputVariable="SendOrder"
6     outputVariable="getResponse"
7     p:PolicyURIs="...policies#RM
8     ...policies#SEC">
```

Exemplo de política em uma composição de serviços.

Proposta de Arquitetura

- Todas as funcionalidades adicionais em uma nova camada.

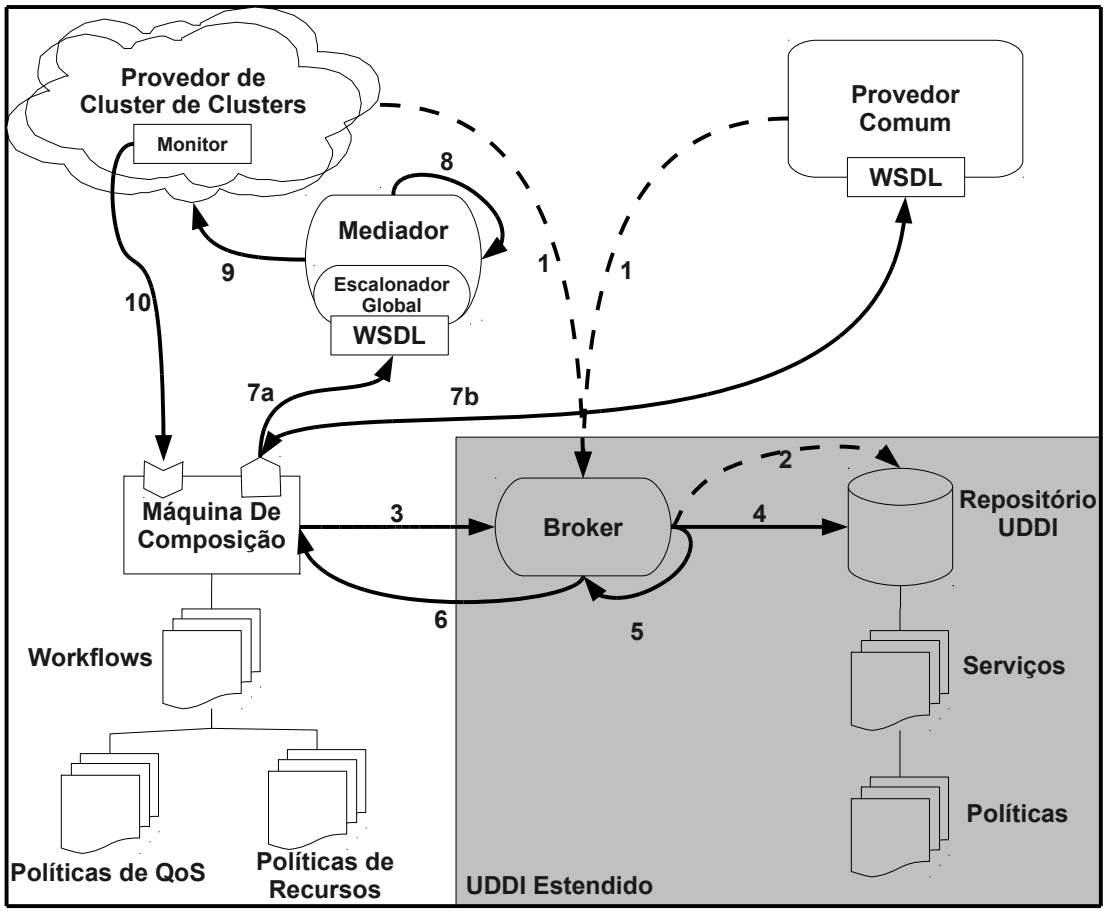


Estudo de caso: SHARCNET

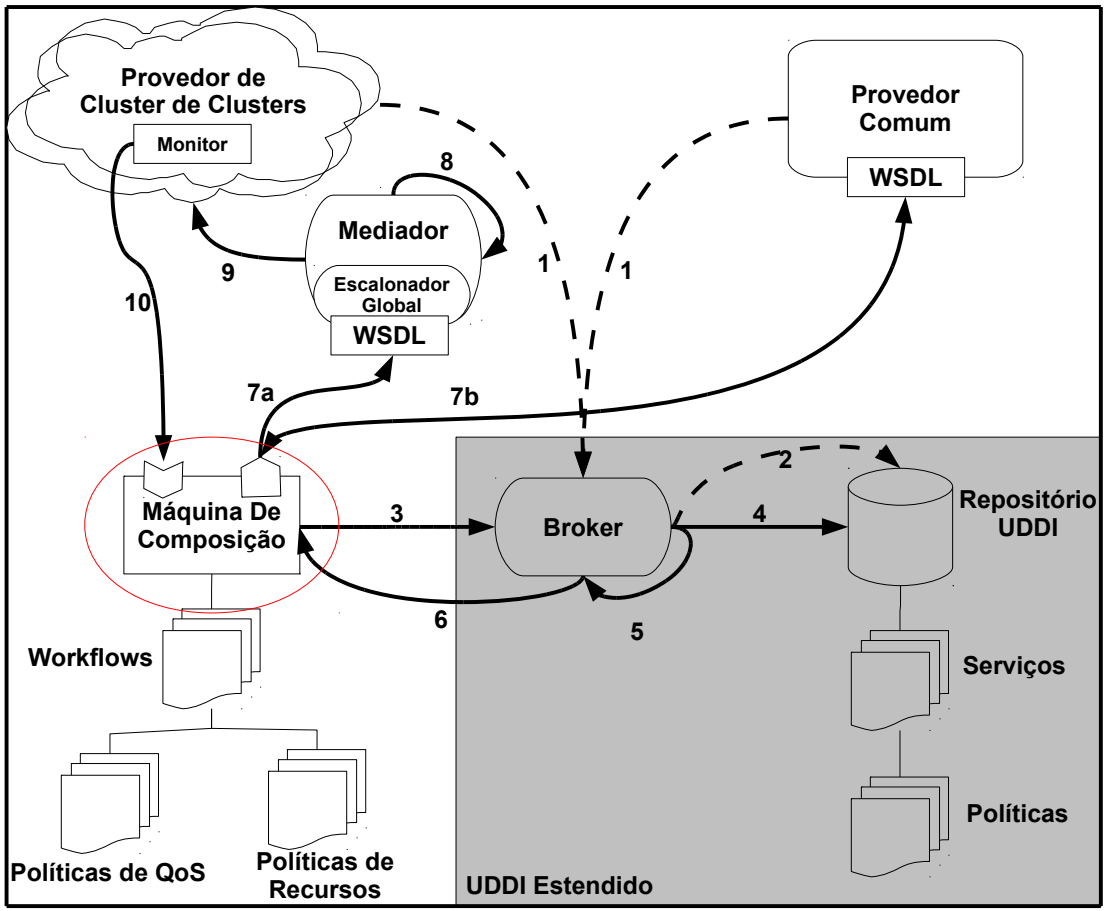
A SHARCNET

- Rede multi-institucional de *clusters* de alta performance.
- Distribuídos em dezesseis instituições acadêmicas na província de Ontário, no Canadá.
- Ausência de escalonador global.
  - "Impossibilidade" de execução automatizada. Workflow!

Arquitetura Aplicada



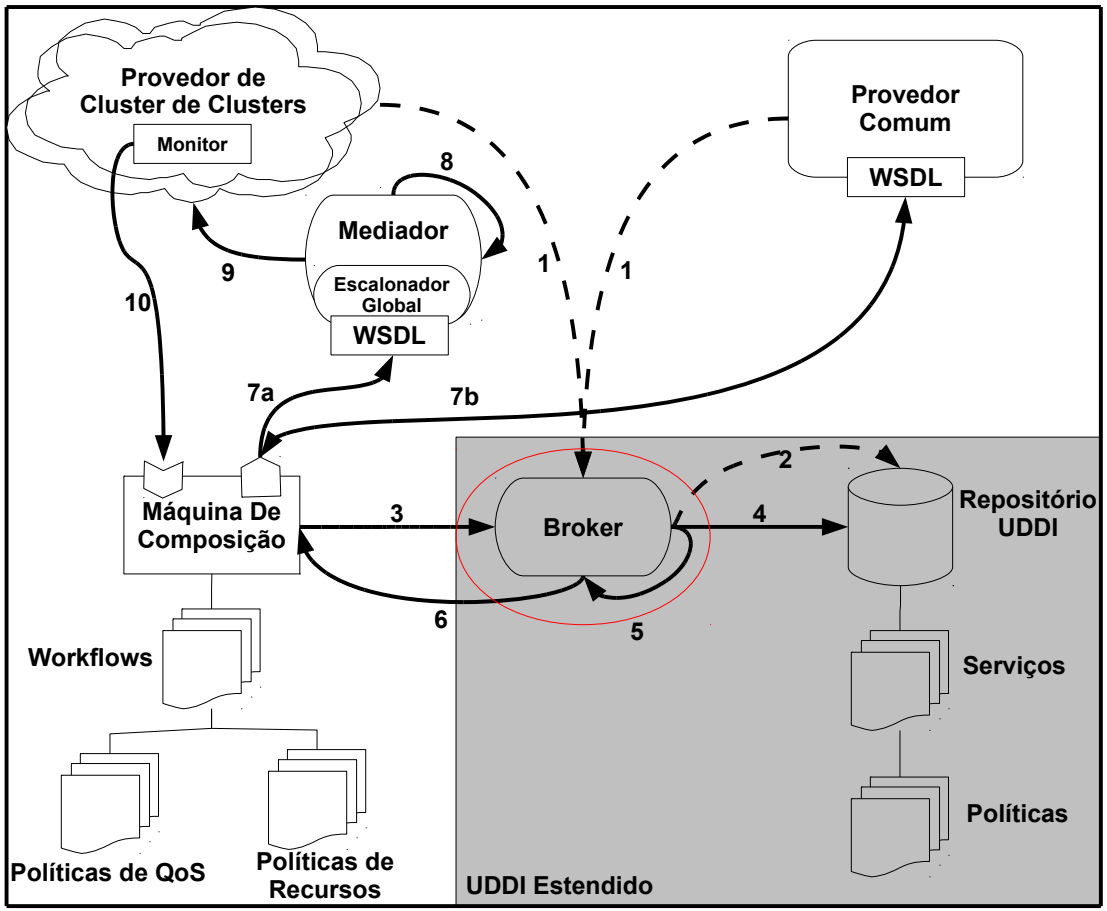
Máquina de Composição



Máquina de Composição

- Componente mais importante para a execução de um processo BPEL.
- *BPEL Service Engine* para Glasfish Open ESB no Netbeans.
- Arquitetura adaptada para funcionalidades padrão do componente.
- Não foi modificada.

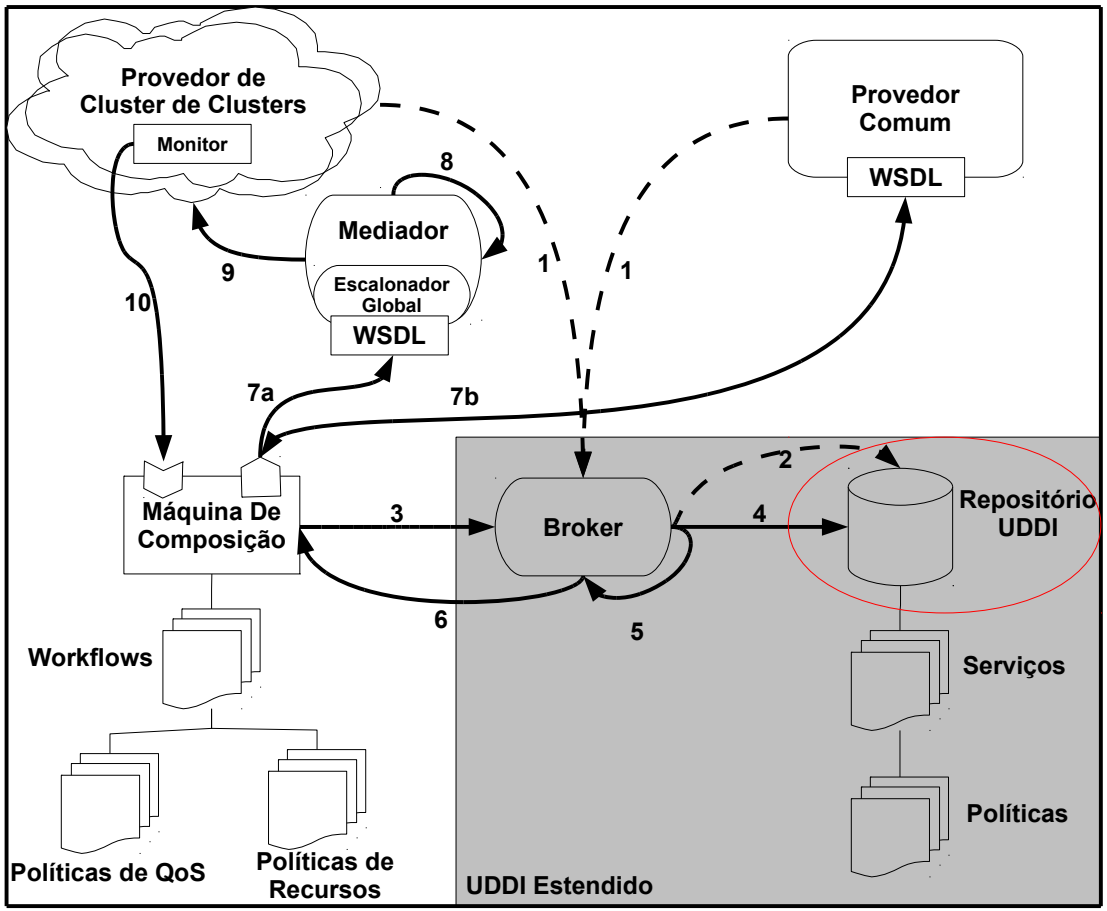
Broker


Provedor de
Cluster de Clusters
Monitor
Provedor
Comum
WSDL
Mediador
Escalonador
Global
WSDL
8
9
10
1
1
7a
7b
Máquina De
Composição
3
Broker
2
4
Repositório
UDDI
6
5
Workflows
Serviços
Políticas
Políticas de QoS
Políticas de
Recursos
UDDI Estendido

Broker

- Parte principal do UDDI estendido proposto por Garcia e Toledo.
- WS de interface para um UDDI.
- Busca por requisitos funcionais e de QoS.
- Retorna o serviço mais indicado.
- *Binding* Dinâmico.

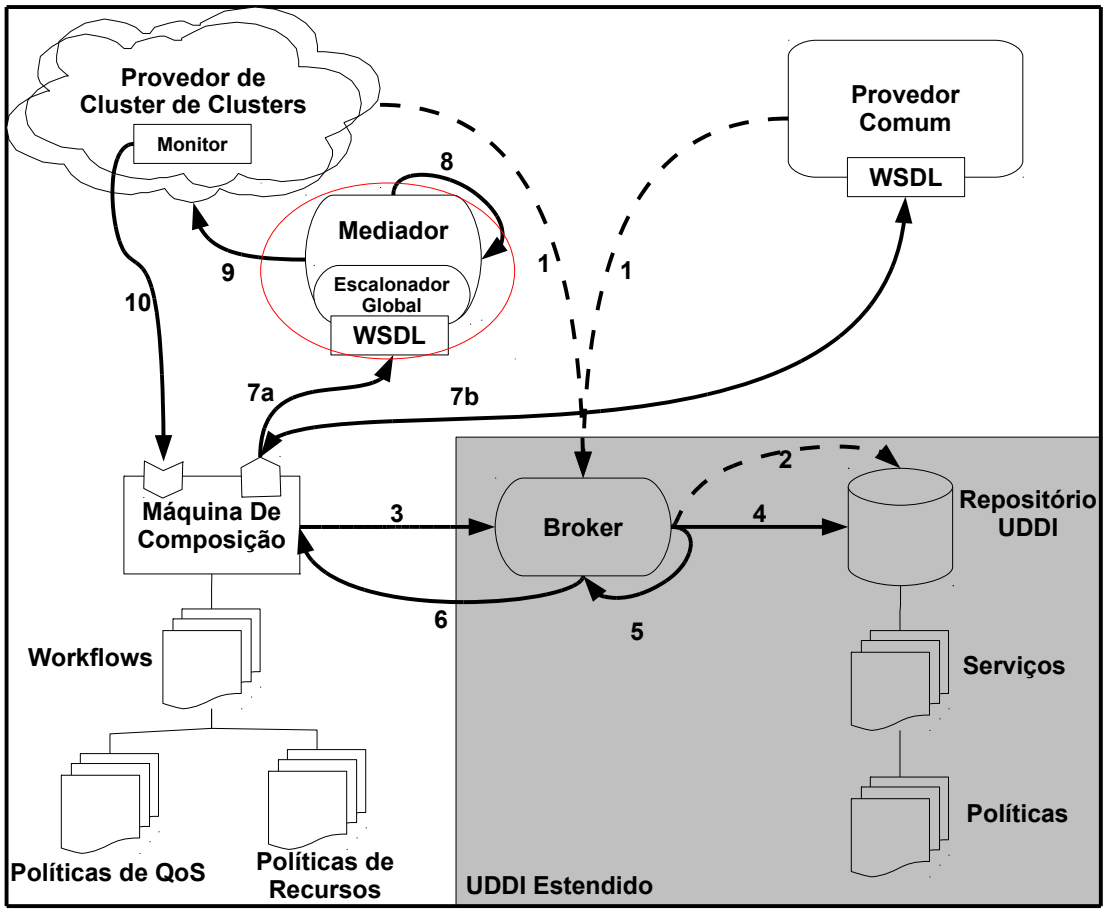
UDDI


Provedor de Cluster de Clusters
Monitor
Provedor Comum
WSDL
Mediador
Escalonador Global
WSDL
8
9
10
1
1
7a
7b
Máquina De Composição
3
Broker
2
4
Repositório UDDI
6
5
Workflows
Serviços
Políticas
Políticas de QoS
Políticas de Recursos
UDDI Estendido

UDDI

- Armazenar os serviços publicados e auxiliar o `Broker` nas buscas.
- JUDDI.

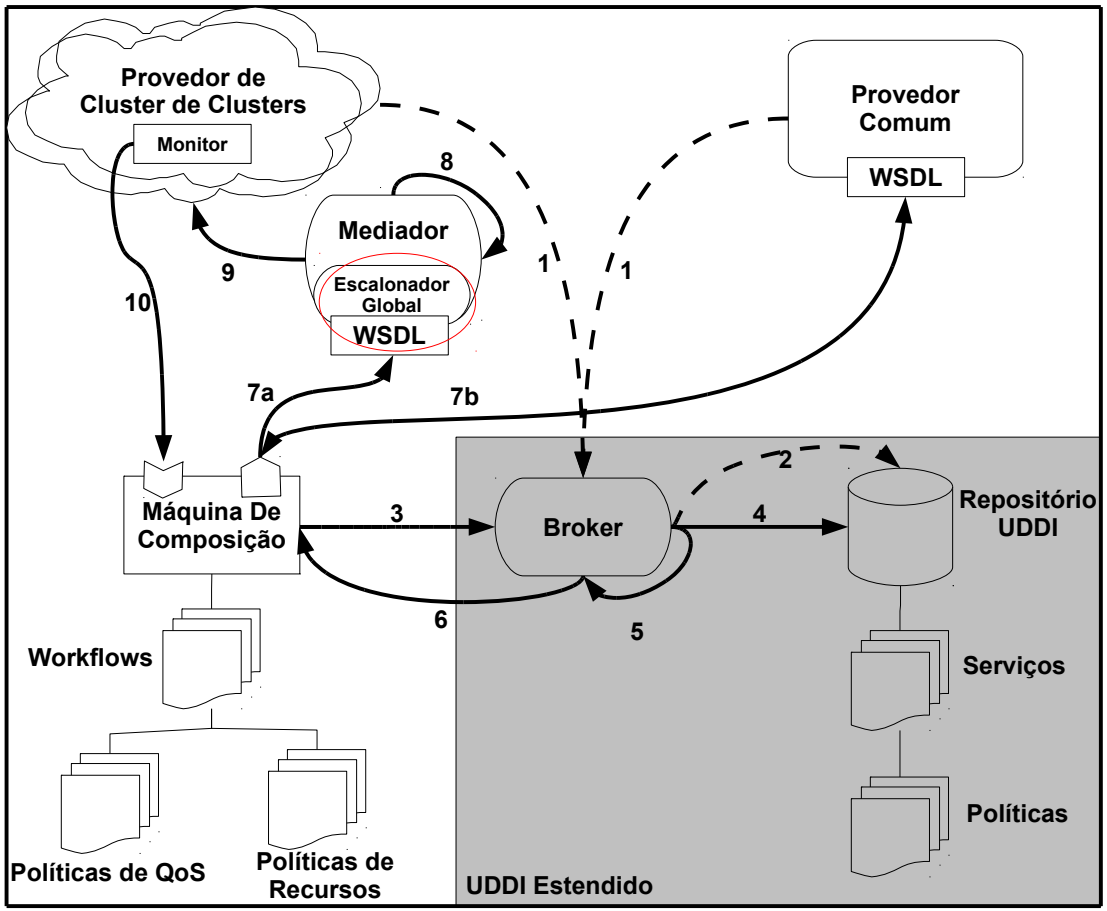
Mediador


Provedor de
Cluster de Clusters
Monitor
Provedor
Comum
WSDL
Mediador
Escalonador
Global
WSDL
8
9
10
1
1
7a
7b
Máquina De
Composição
3
Broker
2
4
Repositório
UDDI
6
5
Workflows
Serviços
Políticas
Políticas de QoS
Políticas de
Recursos
UDDI Estendido

Mediador

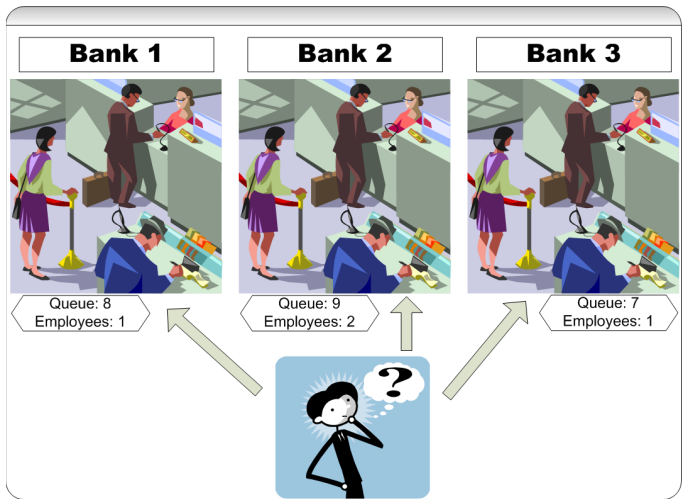
- WS que atua como um *proxy* para a SHARCNET.
- Verifica os requisitos (descritos nas políticas) e invoca os outros componentes da arquitetura.
- Não é centralizado.
- Invocação assíncrona.

Escalonador Global



Escalonador Global

- Parceria com a UWO.
- Compatibilidade com a seleção manual de recursos: Aplicações existentes não são modificadas.
- Cenário:



Fila de Banco

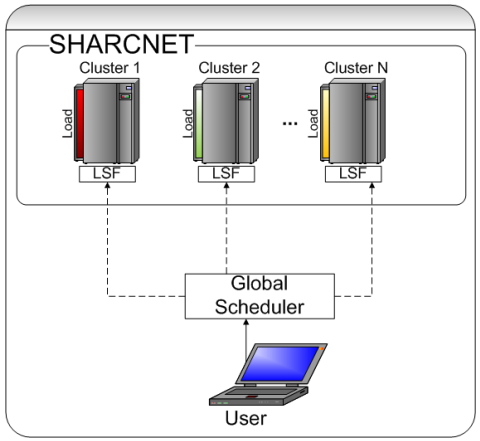
Escalonador Global

- Métrica proposta para comparar a carga de cada HPCC:

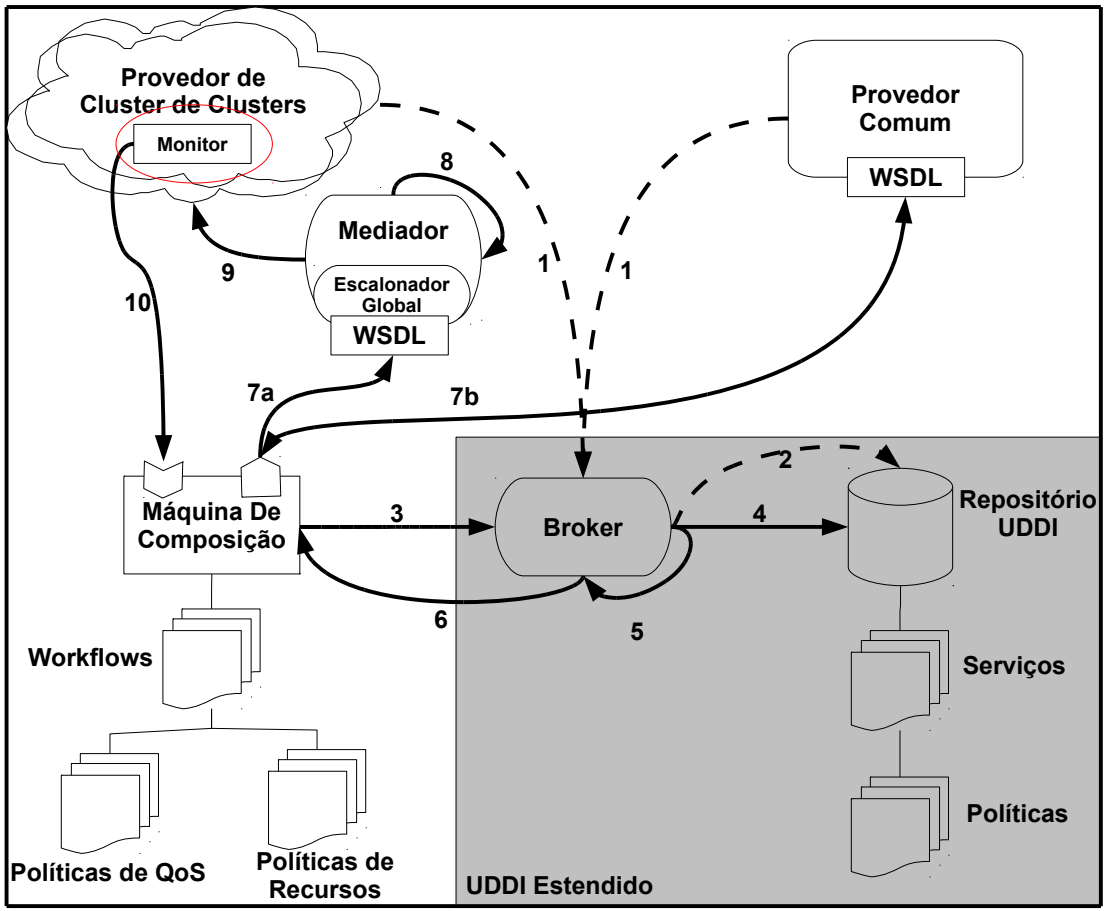
Formula

$$\frac{\textit{Quantidade de processadores requisitados pela fila}}{\textit{Quantidade total de processadores} * \textit{Potencia dos processadores}}$$

- Envia a tarefa para o *cluster* que melhor atende os requisitos.
- Método simples e distribuído.



Monitor


Provedor de Cluster de Clusters
Monitor
Provedor Comum
WSDL
Mediador
Escalonador Global
WSDL
8
9
10
1
1
7a
7b
Máquina De Composição
3
Broker
2
4
Repositório UDDI
6
5
Workflows
Serviços
Políticas
Políticas de QoS
Políticas de Recursos
UDDI Estendido

Monitor

- Monitora o estado dos serviços sendo executados na SHARCNET.
- Uma instância por tarefa.
- Quando a tarefa é concluída, lê o resultado e envia a resposta para o endereço de *callback* (WS-Addressing).

Aplicação

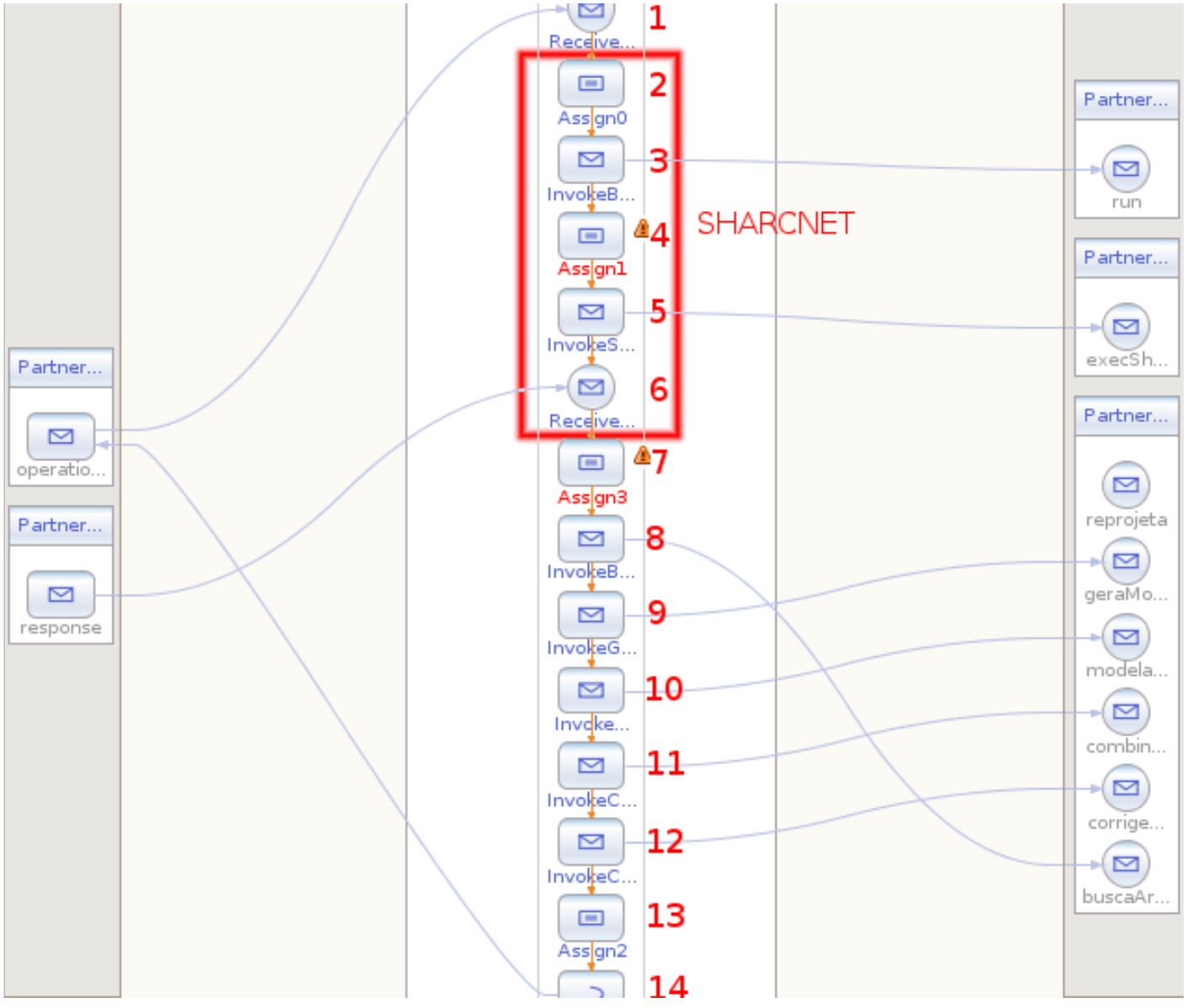
Montage

- Conjunto de ferramentas criadas pela NASA para gerar mosaicos personalizados do céu (WF).
- Entrada FITS (Sistema de Transporte Imagem Flexível).
- Reprojetar imagens, gerar mosaico inicial, modelar planos de fundo, combinar planos de fundo, e gerar mosaico corrigido.
- Dez imagens do Atlas 2MASS da galáxia Pinwheel, ou M101.

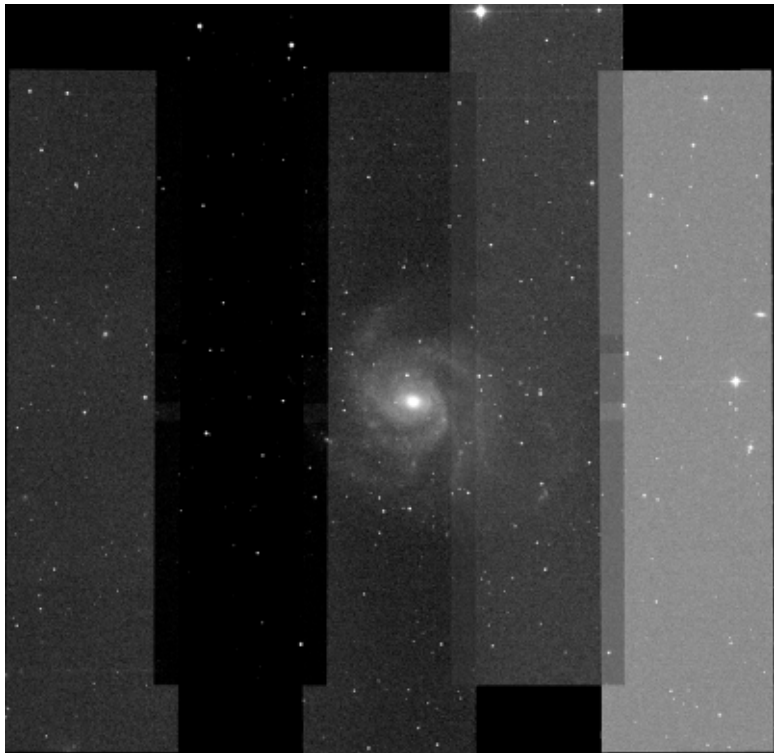
Implementação

- Seis Serviços Web:
  - Um para cada etapa do algoritmo;
  - Extra para transferência de dados.
- Interação descrita com WS-BPEL.
- Etapa 1 (Reprojeção) executada na SHARCNET.

BPEL


Partner...
operatio...
Partner...
response
Receive...
1
Assign0
2
InvokeB...
3
Assign1
4
InvokeS...
5
Receive...
6
Assign3
7
InvokeB...
8
InvokeG...
9
Invoke...
10
InvokeC...
11
InvokeC...
12
Assign2
13
14
SHARCNET
Partner...
run
Partner...
execSh...
Partner...
reprojeta
geraMo...
modela...
combin...
corrige...
buscaAr...

Resultado



Mosaico M101 não corrigido.

Mosaico M101 corrigido.

Conclusões

Conclusões

- Abordagem do problema com WS-BPEL.
- Foco na estrutura de CoC.
- Solução implementada de ponta a ponta se mostra viável.
  - Processo muitas vezes lento.
  - Complexidade para utilizar funcionalidades que não são comuns.
  - Dependente da máquina de execução utilizada.

Principais Contribuições

- As principais contribuições incluem:
  - Extensão à WS-BPEL para a especificação de *workflows* em CoCs (WS-Policy);
  - Uma proposta de arquitetura para a execução de *workflows* em um *cluster* de *clusters*;
  - Um estudo de caso aplicando essa arquitetura na rede SHARCNET;
  - Uma implementação de escalonador global para a SHARCNET;
  - Especificação e implementação de um *workflow* científico real, utilizando o ambiente criado.

Trabalhos Futuros

- Gerenciamento de dados e transferência de informações.
- Tolerância a falhas.
- Ferramenta para auxiliar o monitoramento do estado de cada *workflow* em execução.

Publicações

- Posters e artigos resumidos:
  - "*A global scheduler for SHARCNET*"; *SHARCNET Research Day 2009*; Waterloo, Ontário, Canadá.
  - "*Estendendo a linguagem WS-BPEL para a execução de workflows em um cluster de clusters*"; 4o Workshop de Teses de Doutorado UNICAMP, 2009; Campinas, Brasil.

Publicações

- Artigos completos:
  - "*An Infrastructure for Executing WS-BPEL Workflows in a Cluster of Clusters*"; *IEEE symposium on Computers and Communications (ISCC 2010)*; Riccione, Itália.

Palestras Aguardando Avaliação

- "*Orquestrando Serviços Web em Java com WS-BPEL*"; *JavaOne: Oracle OpenWorld Latin America 2010*; São Paulo, Brasil.
- "*Implementando Serviços Web Assíncronos em Java*"; *JavaOne: Oracle OpenWorld Latin America 2010*; São Paulo, Brasil.
- "*Orquestração de Serviços com WS-BPEL*"; *Latinoware 2010*; Foz do Iguaçu, Brasil.

Agradecimentos

- Prof. Maria Beatriz.
- Prof. Miriam Capretz (UWO).
- Membros da Banca.
- Instituto de Computação.
- FAPESP.

Obrigado!

Dúvidas?

Uma Infraestrutura Para a Execução de Workflows WS-BPEL em Clusters de Clusters

Thiago Alvarenga Lechuga

21/10/2010

Orientadora: Maria Beatriz Felgar de Toledo
Agência Financiadora: FAPESP

Backup

Sumário

SHARCNET - HW

- Three primary clusters that have already been listed in the top500 containing 267 nodes with 1068 processors, 768 nodes with 1536 processors and 768 nodes with 3072 processors
- Each of these clusters provides 70TB of high speed storage
- 17 secondary clusters, ranging from 32 to 64 nodes and totaling more than 8,000 processors and 200TB of storage
- Interconnection between the nodes in each cluster includes G2 Myrinet, Quadrics (Elan, Elan 4 and 3) and Gigabit Ethernet
- Communication between clusters is performed by a dedicated high-speed network and a 10 Gigabit Ethernet

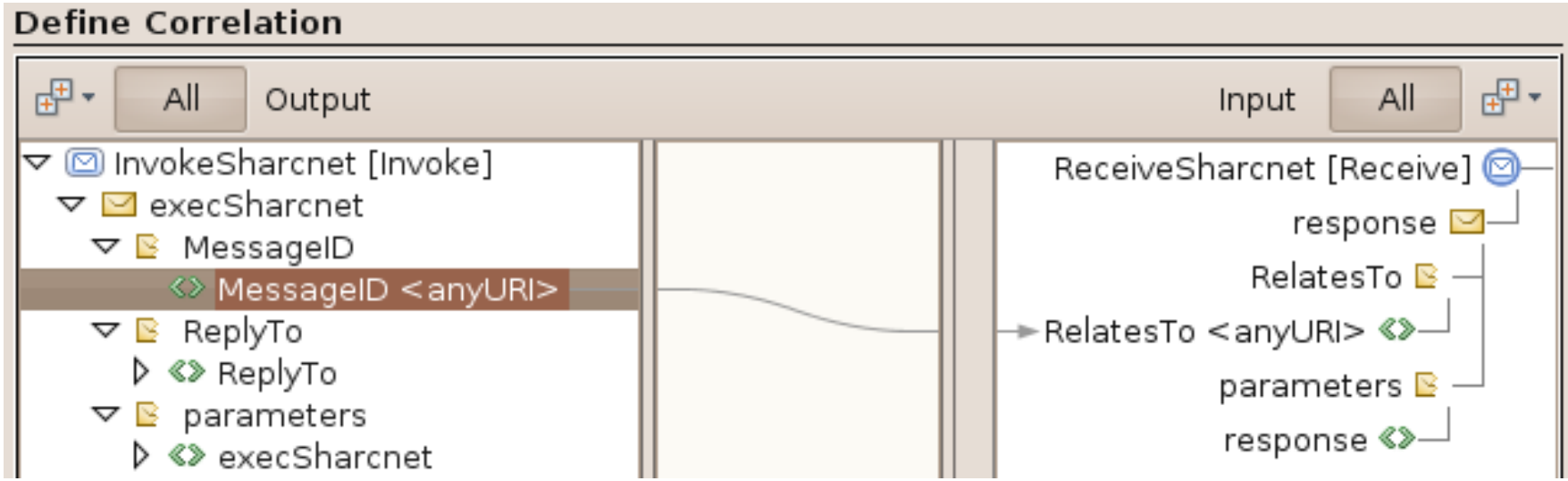
SHARCNET - SW

- Studies from many areas that traditionally require high performance computing: chemistry, physics, material science, and engineering; and newer areas: business, economics and biology
- Operating systems are variants of Linux
- Job Scheduling is performed by the Load Sharing Facility (LSF)
- Parallel computing can be developed using MPI or OpenMPI
- Each cluster has a master node that acts as a local scheduler

WS-PolicyAttachment

WS-PolicyAttachment defines the attribute of a PolicyURI, which connects a WS-Policy to an XML element. Accordingly, these attributes allow policies to be linked to WS-BPEL elements that represent service compositions or specific services in compositions.

Correlação



Estrutura de `Correlation` sendo criada.

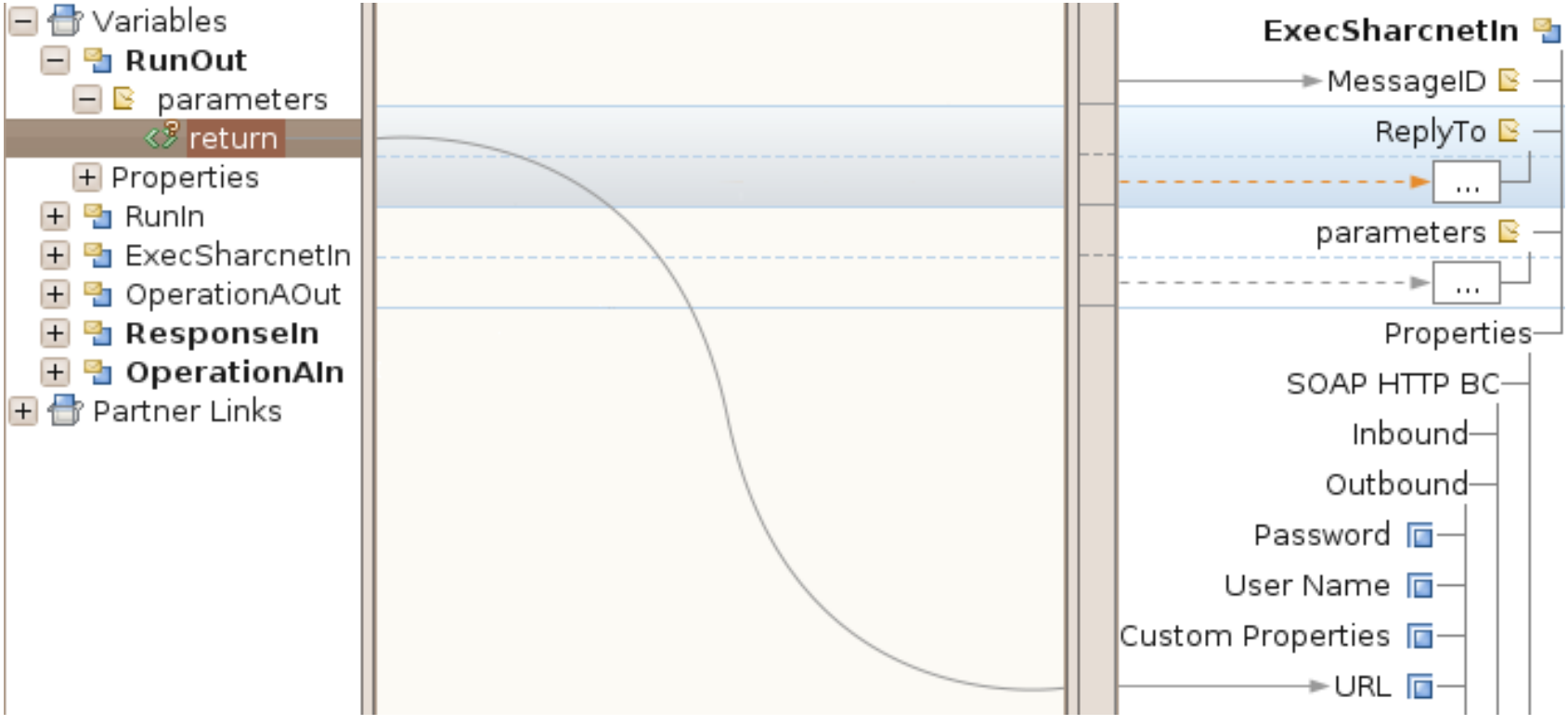
Binding Dinâmico

O parâmetro `return`, da variável que recebe a saída do `Broker`, é atribuído ao parâmetro URL, das propriedades SOAP do serviço `ExecSharcnetIn`.

```
1 <assign name="Assign1">
2     <copy>
3         <from>$RunOut.parameters/return</from>
4         <to variable="ExecSharcnetIn" sxnmp:nmProperty="org.glassfish.openesb.
              outbound.address.url" />
5     </copy>
6 </assign>
```

Configurando um *binding* dinâmico.

Binding Dinâmico



Configurando um *binding* dinâmico no Netbeans.

Related Work

- Most authors agree that WS-BPEL can and should be adopted for grid service composition with some adjustments
- Many studies [5, 7, 8, 11, 12], consider the execution of workflows in OGSI (Open Grid Services Infrastructure) and WSRF grids
- Ezenwoye et al. [5], the authors focus on using BPEL to run WSRF services that implement the factory pattern
- Ma et al. [8] choose not to extend WS-BPEL, but to modify the ActiveBPEL engine instead
- Emmerich et al. [4] use specifically GT3 and Cybok [3] uses Condor structures in their respective works

Related Work - Cont.

- Takpe et al. [9], as well as Qin and Bauer [10] demonstrate scheduling methods in a cluster of clusters
- Thus far, most of the proposals have been limited to formal proofs and/or simulations without implementation in a real environment
- Compatibility problem: This simple fact invalidates some proposals [1, 2], which assume all requests are directed through the new "global scheduler"
- Hunold et al. [6] also propose a scheduling method for clusters using the technique of postponing: Starvation

Marta Beltrán and Antonio Guzmán.

How to balance the load on heterogeneous clusters.
Int. J. High Perform. Comput. Appl., 23(1):99–118, 2009.

Siu-Cheung Chau and Ada Wai-Chee Fu.

Load balancing between computing clusters.
In *Proceedings of the Fourth International Conference on Parallel and Distributed Computing, Applications and Technologies, 2003. PDCAT'2003*, pages 548–551, Aug. 2003.

Dieter Cybok.

A grid workflow infrastructure.
Concurr. Comput. : Pract. Exper., 18(10):1243–1254, 2006.

W. Emmerich, B. Butchart, L. Chen, B. Wassermann, and S.L. Price.

Grid service orchestration using the business process execution language(BPEL).
Journal of Grid Computing, 3(3-4):283–304, 2005.

Onyeka Ezenwoye, S. Masoud Sadjadi, Ariel Cary, and Michael Robinson.

Grid service composition in BPEL for scientific applications.
GADA, 2007.

S. Hunold, T. Rauber, and F. Suter.

Scheduling dynamic workflows onto clusters of clusters using postponing.
In *8th IEEE International Symposium on Cluster Computing and the Grid. CCGRID '08*, pages 669–67[illegible]
May 2008.

Frank Leymann.

Choreography for the grid: towards fitting BPEL to the resource framework.

Concurr. Comput. : Pract. Exper., 18(10):1201–1217, 2006.

Ru-Yue Ma, Yong-Wei Wu, Xiang-Xu Meng, Shi-Jun Liu, and Li Pan.
Grid-enabled workflow management system based on BPEL.
Int. J. High Perform. Comput. Appl., 22(3):238–249, 2008.

T. N'Takpe, F. Suter, and H. Casanova.
A comparison of scheduling approaches for mixed-parallel applications on heterogeneous platforms.
In *Sixth International Symposium on Parallel and Distributed Computing. ISPDC '07*, pages 35–35, July 2007.

Jinhui Qin and M.A. Bauer.
An improved job co-allocation strategy in multiple hpc clusters.
In *21st International Symposium on High Performance Computing Systems and Applications, 2007. HPCS 2007*, pages 18–18, May 2007.

Aleksander Slomiski.
On using BPEL extensibility to implement OGSI and WSRF grid workflows.
Concurr. Comput. : Pract. Exper., 18(10):1229–1241, 2006.

A. Tsalgatidou, G. Athanasopoulos, M. Pantazoglou, C. Pautasso, T. Heinis, R. Gronmo, Hjordis Hoff, Arne-Jorgen Berre, M. Glittum, and S. Topouzidou.
Developing scientific workflows from heterogeneous services.
SIGMOD Rec., 35(2):22–28, 2006.