

# Workflows Científicos em Plataformas de Computação de Alto Desempenho

**Kelly R. Braghetto**

Departamento de Ciência da Computação  
Instituto de Matemática e Estatística  
Universidade de São Paulo

AGA0513 – eScience em Astronomia  
2 de junho 2017

## Sobre mim

- Professora do Departamento de Ciência da Computação do IME-USP
- Mestrado (2006) e doutorado (2011) em Ciência da Computação pela USP
- Desde 2003 desenvolve pesquisas na área e Modelagem de Dados e Processos
  - Gerenciamento de workflows científicos e de negócio,
  - Modelagem formal de processos, avaliação de desempenho de processos via modelagem analítica
  - Bancos de dados

## 1 Introdução

## 2 Modelagem de Workflows Científicos

- Terminologia básica
- Fluxo de controle  $\times$  fluxo de dados

## 3 Execução de Workflows Científicos

- Recursos

## 4 Sistemas de Gerenciamento de Workflows Científicos

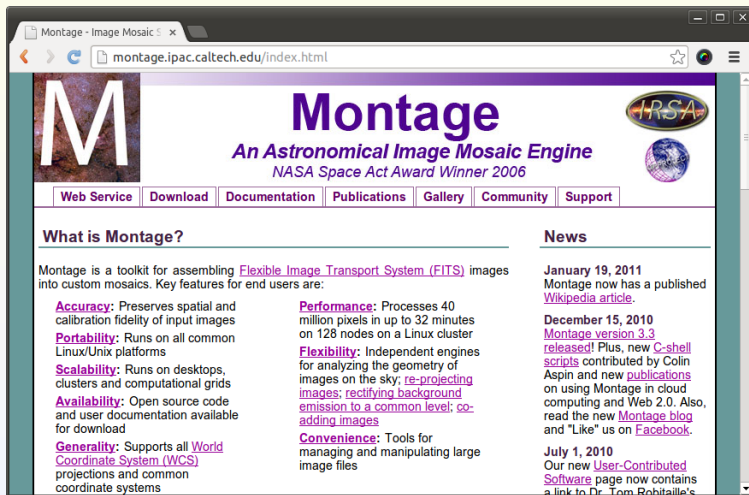
- Taverna
- Kepler
- Pegasus

# Introdução

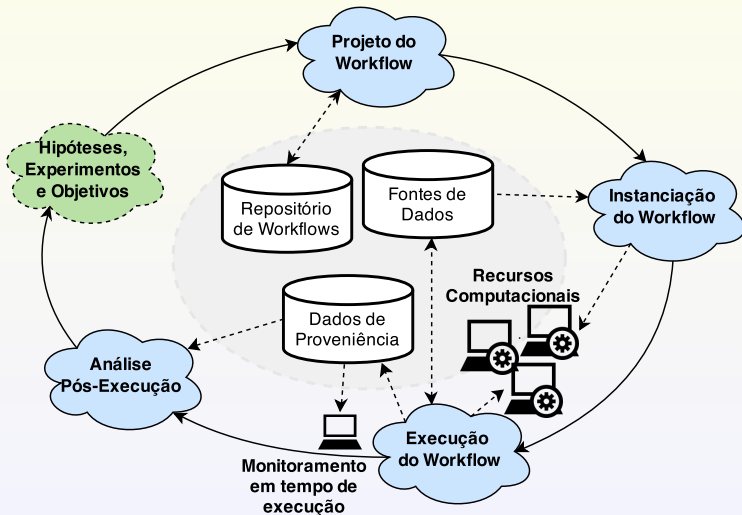
# Workflows Científicos

- São automações de experimentos ou de processos científicos, expressas em termos das atividades a serem executadas e, principalmente, das dependências dos dados manipulados
- Tendem a ser computacionalmente intensivos
- São fortemente voltados à manipulação de grandes volumes de dados complexos

## Exemplo famoso



## Ciclo de vida dos workflows científicos



# O projeto de workflows científicos

- Consiste na construção de um modelo para o workflow
- Pode ser feito por meio de diferentes linguagens ou modelos de representação
- Pode se embasar em outros workflows já existentes
- É conveniente de se representar em formatos digitais e armazenar em repositórios:
  - reuso, compartilhamento com outros pesquisadores, etc.

## A instanciação de workflows científicos

- Corresponde à preparação necessária para uma execução particular do workflow:
  - Cada nova execução requer dados de entrada e parâmetros de configuração definidos pelos cientistas
- Pode envolver a seleção e alocação dos recursos computacionais para a execução
- Cuida da transferência dos dados de entrada para os computadores onde a execução será realizada

## A execução de workflows científicos

- Ocorre nos recursos computacionais disponíveis:
  - Atribui-se um recurso para cada atividade
  - Atividades processam dados de entrada e produzem novos dados, que podem ser usados por outras atividades
- Pode registrar dados intermediários gerados na instância e suas *informações de proveniência*
  - Dados de entrada e os parâmetros de configuração
  - Registro das atividades já executadas, seus tempos de início e término, os recursos usados, etc.
  - Referências para os dados de entrada e saída de cada atividade

Intenções: avaliação dos resultados; recuperação em caso de falhas; reprodutibilidade dos experimentos

# A análise pós-execução de workflows científicos

- Corresponde à inspeção e interpretação dos resultados obtidos
- Pode conduzir a uma revisão da hipótese ou do objetivo experimental inicial, ou ainda à identificação de ineficiências da execução
  - Reprojetado do workflow  $\Rightarrow$  nova iteração do seu ciclo de vida

# Modelagem de Workflows Científicos

# Terminologia básica sobre representação de workflows

**Modelo** define as atividades a serem realizadas no workflow para que um objetivo científico seja alcançado

**Atividade** é uma unidade atômica (= indivisível) de trabalho em um modelo de workflow

**Conector** é usado para definir alguma relação de precedência ou algum tipo de transferência de dados entre atividades

**Instância** é uma execução particular (= instanciação) de um modelo de workflow

## Modelo de workflow

Define as atividades a serem realizadas no workflow para que um objetivo científico seja alcançado

- Estabelece a ordem (ainda que parcial) na qual as atividades do workflow devem ser executadas
- É comumente representado por meio de um **arcabouço formal** ou de uma **linguagem de especificação de workflows**
  - arcabouços formais possibilitam a análise de propriedades dos workflows que neles são modelados
  - linguagens específicas de domínio permitem a especificação de modelos em um nível de detalhamento mais próximo ao requerido para a execução dos workflows

## Perspectivas para a modelagem de workflows

**Fluxo de controle** descreve a ordem de execução, definindo a transferência de controle entre atividades conectadas e habilitando uma atividade para execução quando as atividades que a precedem são concluídas

**Fluxo de dados** descreve as transferências de dados entre as atividades, estabelecendo as dependências das atividades com relação aos dados que elas manipulam

**Organizacional** atrela ao workflow uma estrutura organizacional, por meio da definição de papéis (desempenhados por pessoas ou equipamentos) responsáveis pela execução das atividades

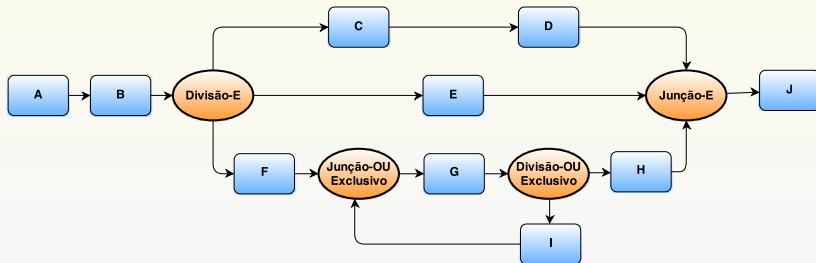
**Tratamento de exceções** lida com as causas das exceções e as ações que precisam ser tomadas nos seus tratamentos

# Perspectivas para a modelagem de workflows

## Fluxos de dados

- Em workflows científicos, a predominância é da modelagem de fluxos de dados
  - Aplicações científicas são intensivas em dados
- Cientistas rotineiramente já descrevem seus experimentos científicos em termos de fluxos de dados
  - coleta de dados brutos → filtragem → transformação → análise → obtenção de dados derivados que sustentam hipóteses científicas

## Estruturas para representação de fluxos de controle



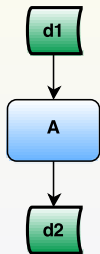
Exemplo de workflow modelado usando as construções básicas de fluxo de controle

# Estruturas para representação de fluxos de dados

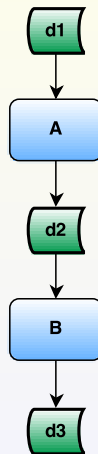
## Construções básicas:

- **Processamento**: é realizado por uma atividade sobre dados de entrada para produzir dados de saída
- **Pipeline**: combina um ou mais processamentos sequencialmente, de forma que cada atividade processe os dados produzidos pela atividade que a precedeu e que a sua saída seja usada como entrada para a próxima atividade

## Estruturas para representação de fluxos de dados



Processamento



*Pipeline*

## Estruturas para representação de fluxos de dados

### Construções básicas:

- ***Distribuição de dados***: é feita por uma atividade que produz dois ou mais conjuntos de dados de saída que são recebidos como entrada por duas ou mais atividades

Pode ser feita para dois propósitos distintos:

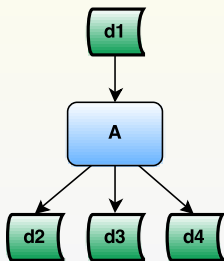
- produzir novos conjuntos de dados para serem consumidos por múltiplas atividades; ou
- particionar um grande conjunto de dados recebido como entrada em subconjuntos menores, para que esses possam ser processados (paralelamente) por outras atividades no workflow

# Estruturas para representação de fluxos de dados

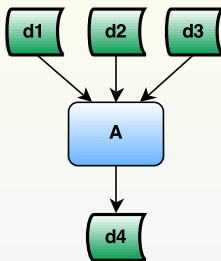
## Construções básicas:

- **Agregação de dados:** é feita por uma atividade que agrega e processa dados de saída de duas ou mais atividades, gerando uma combinação dos dados como saída
  - Pode resultar em sincronização de linhas de execução paralelas
- **Redistribuição de dados:** é feita por uma atividade que combina a função de agregação de dados à função de distribuição dos dados
  - a atividade pode tanto sincronizar linhas de execução paralelas já existentes quanto criar novas linhas de execução paralelas

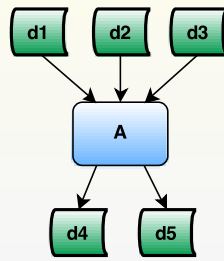
## Estruturas para representação de fluxos de dados



Distribuição  
de dados

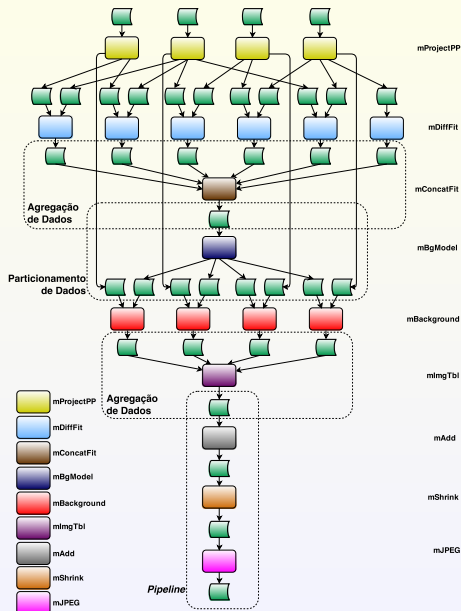


Agregação  
de dados

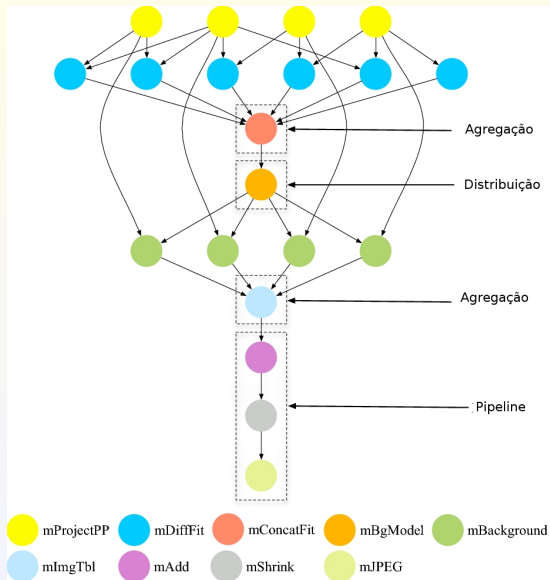


Redistribuição  
de dados

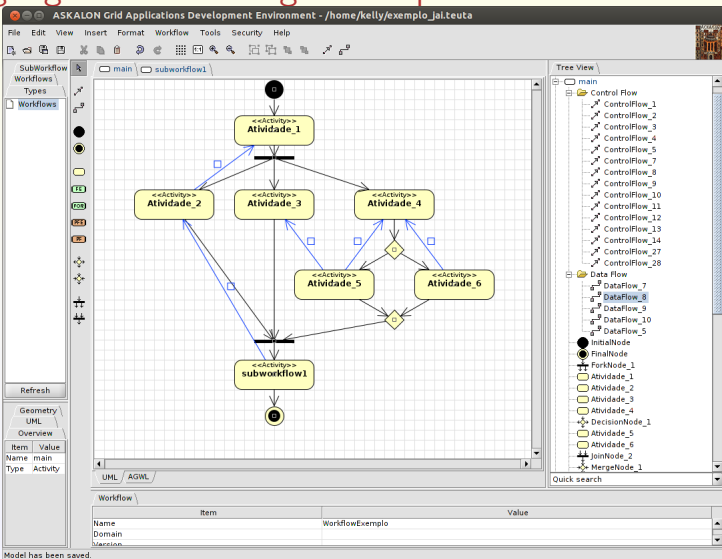
Fluxo de dados  
no workflow  
**Montage**  
[http://montage.  
ipac.caltech.edu/](http://montage.ipac.caltech.edu/)



## Workflow Montage (sem a visualização dos itens de dados)

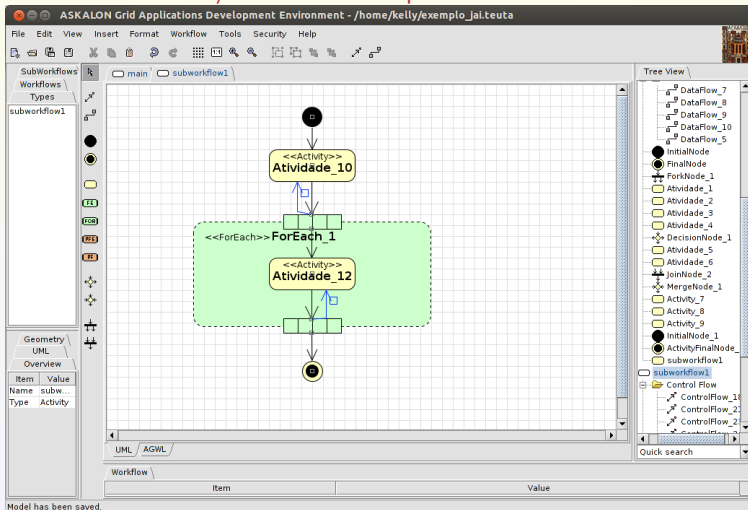


# As linguagens de modelagem “na prática”



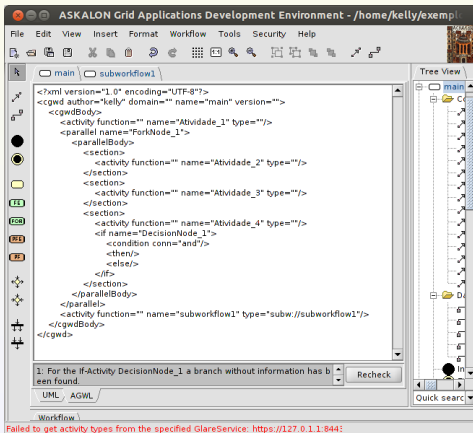
# As linguagens de modelagem “na prática”

Composicionalidade: construção de workflow a partir de subworkflows



# As linguagens de modelagem “na prática”

## Representação textual do modelo



# Execução de Workflows Científicos

# Escalonamento

## Definição informal

Escalonamento se refere a uma alocação, no tempo, de um conjunto limitado de recursos a tarefas. É um processo de decisão que tem como intuito otimizar um ou mais objetivos.

## Exemplos

**Recursos** máquinas em uma fábrica; pistas em um aeroporto; CPUs em um computador

**Tarefas** operações de produção; pousos e decolagens; programas de computadores

**Objetivos** otimizar o tempo do término da última tarefa; minimizar o número de tarefas “atrasadas”

## Recursos usados na execução de workflows

A execução de workflows intensivos em dados e processamento dependem de plataformas de computação de alto desempenho.

Cientistas têm à disposição três tipos de plataformas:

- Aglomerados (*clusters*) de computadores
- Grades de computação
- Plataformas de computação em nuvem

## Aglomerados de computadores

Um conjunto de computadores (independentes) que trabalham juntos como se fossem um só. Grosso modo<sup>1</sup>, têm as seguintes características:

- Hardware parecido (em geral, consideramos os recursos uniformes)
- Fisicamente próximos
- Interligados por uma rede

---

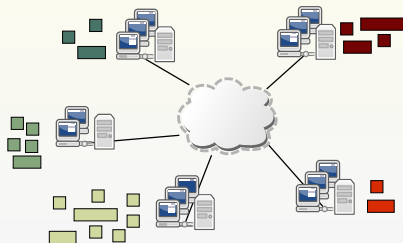
<sup>1</sup>Não existe uma definição precisa para o que é um aglomerado.

## Grades de computação

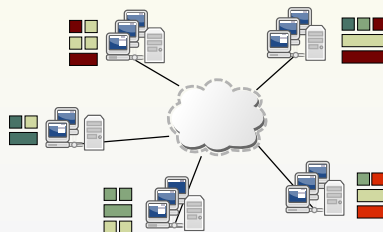
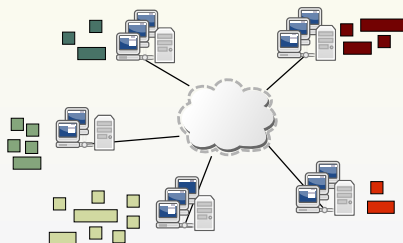
Conjunto de computadores e aglomerados que podem ser utilizados por qualquer usuário. A ideia é que qualquer um possa utilizar os recursos computacionais ociosos dos outros. Tipicamente:

- Os recursos estão dispersos geograficamente
- Os recursos são heterogêneos
- Computação disponível como um serviço, sem a preocupação de onde o serviço é oferecido. Analogia com a rede elétrica (de qual usina hidrelétrica vem a eletricidade desta tomada?)

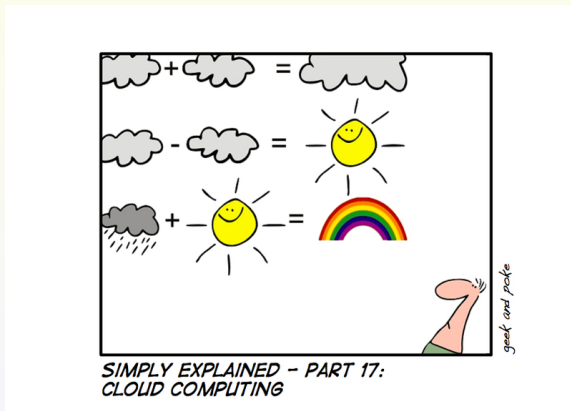
## Grades de computação



## Grades de computação



## Plataformas de computação em nuvem



**Figura:** O que é computação em nuvem?

# Plataformas de computação em nuvem

*É um modelo que possibilita **acesso ubíquo** (via internet), de forma conveniente e **sob demanda** a um conjunto de recursos de computação configuráveis (por exemplo, redes, servidores, dispositivos de armazenamento, aplicações e outros serviços) **que podem ser rapidamente provisionados e dispensados** com o mínimo esforço de gestão ou interação do prestador de serviço.*

Instituto Nacional de Padrões e Tecnologia dos EUA (NIST):  
<http://csrc.nist.gov/publications/PubsSPs.html#800-145>

# Plataformas de computação em nuvem

## Características:

- Recursos *quasi* ilimitados
- Podem ser alugados ou devolvidos a qualquer momento
- Difícil prever o desempenho dos recursos
- Investimento em capital vs. custos operacionais

# Plataformas de Computação em Nuvem

## Classificação quanto à estrutura

- *Privadas*: recursos computacionais próprios (ex.: Nuvem USP)
- *Comunitárias*: recursos de várias organizações
- *Públicas*: recursos de fornecedores comerciais (ex.: *Amazon Elastic Compute Cloud* <sup>2</sup> e *Microsoft Azure Services* <sup>3</sup>);
- *Híbridas*: combinações das estruturas anteriores.

---

<sup>2</sup>Amazon AWS: <http://aws.amazon.com>

<sup>3</sup>Microsoft Azure: <http://azure.microsoft.com>

# Sistemas de Gerenciamento de Workflows Científicos

# Sistema de gerenciamento de workflows científicos (SGWC)

## Definição

É um sistema que permite definir, modificar, gerenciar, executar e monitorar workflows científicos, utilizando um nível de abstração que oculta dos cientistas as technicalidades envolvidas nessas tarefas.

## Funcionalidades oferecidas pelos SGWCs

- Interface de usuário personalizável e interativa
- Ferramentas de apoio à reprodutibilidade do experimento
- Integração a ferramentas e serviços distribuídos heterogêneos
- Apoio à execução em sistemas computacionais de alto desempenho
- Monitoramento da execução de um workflow e tratamento de falhas na execução

# Subsistemas que compõem os SGWCs

- Subsistema de projeto
- Subsistema de visualização e apresentação
- Subsistema de execução
- Subsistema de gerenciamento de atividades
- Subsistema de gerenciamento de dados
- Subsistema de gerenciamento de proveniência
- Subsistema de monitoramento

## Subsistema de projeto de workflows

- Permite a criação e a modificação de especificações (modelos)
- Produz especificações em formato digital, descritas em uma linguagem de modelagem de workflows
- Em alguns casos, estabelece a interoperabilidade entre diferentes SGWCs
  - pode usar linguagens de representação padronizadas

# Subsistema de apresentação e representação

- Permite visualizar o workflow e os dados produzidos por ele em diferentes formatos
- Desejável:
  - uma apresentação que facilite a análise dos dados e de sua proveniência
  - possibilite a extração de conhecimento a partir do grande volume de dados e metadados gerados pela execução de diversos experimentos

## Subsistema de execução de workflows

- Gerencia a execução dos workflows
- Cria e executa instâncias de workflows
- Ampara-se no modelo digital dos workflows
- Escalona as atividades nos recursos computacionais disponíveis
- Controla a execução concorrente de múltiplas instâncias
- Faz uso do *subsistema de gerenciamento de atividades* para lidar a possível heterogeneidade do sistema distribuído
- Faz uso do *subsistema de gerenciamento de dados* para gerenciar os dados usados e produzidos pelas atividades e lidar com problemas como o da heterogeneidade e grande volume dos dados

# Subsistema de gerenciamento de proveniência

## Subsistema de monitoramento

Responsáveis por:

- Representação, armazenamento e consulta dos dados e metadados de proveniência
- Monitoramento do status da execução das instâncias, detectando falhas e auxiliando no seu tratamento

## Alguns SGWCs “famosos”

**ASKALON** <http://www.askalon.org/>

**Kepler** <http://kepler-project.org/>

**Pegasus** <http://pegasus.isi.edu/>

**Taverna** <http://www.taverna.org.uk/>

**Triana** <http://www.trianacode.org/>

**SciCumulus** <http://sourceforge.net/projects/scicumulus/>

E alguns sistemas “famosos” que não são SGWCs “clássicos”

**Galaxy** <http://galaxyproject.org/>

**Swift** <http://swift-lang.org/>

**VisTrails** <http://www.vistrails.org/>

# Taverna

- Parte do projeto myGrid [<http://www.mygrid.org.uk/>]
- Motivado por problemas de Ciências Biológicas, mas atualmente usado em várias outras áreas (ciências sociais, música, astronomia, etc.)
- Permite a modelagem de workflows a partir de *componentes*, i.e., serviços **remotos** (serviços web) ou **locais** (*scripts*) que podem ser reutilizados

# Taverna

## Modelo de workflow

- Um workflow Taverna é especificado como um grafo direcionado
- Os vértices do grafo (*processadores*) são os serviços que implementam as atividades
- Cada arco conecta um par entrada-saída e denota dependência de dados entre as atividades
- Há também *ligações de controle* que indica que uma atividade só pode terminar depois da execução de outra

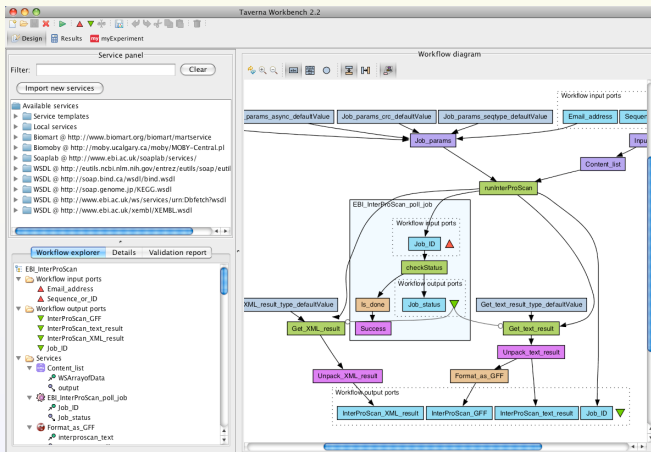
# Taverna

## Escalonamento

- Os serviços executados pelos processadores devem ser implantados manualmente antes da execução do workflow
- Não há necessidade de mapear a execução das atividades aos recursos computacionais
- O Taverna habilita uma atividade assim que ela tiver algum dado para processar. Cada atividade é executada em uma *thread* separada

# Taverna

## Workbench



## myExperiment

- O Taverna está integrado ao portal myExperiment [<http://www.myexperiment.org/>]
- O portal permite que cientistas do mundo inteiro compartilhem seus workflows com outros pesquisadores
- Basta a URL do workflow; o Taverna se encarrega de baixá-lo e executá-lo automaticamente
- No final de maio de 2017, o portal disponibilizava cerca de 3800 workflows

# Kepler

- SGWC de código-aberto desenvolvido por pesquisadores da Universidade da Califórnia (Davis, Santa Barbara e San Diego)
- Uma metáfora de **diretor** / **ator** para representar visualmente os componentes de um workflow
  - os atores especificam *que* processamento é feito
  - o diretor especifica *quando* o processamento ocorre

# Kepler

## Diretor

- Cada modelo de computação é representado por um tipo de diretor
  - o SDF Director executa o workflow de forma síncrona, um componente por vez
  - o PN Director executa todos os componentes em paralelo
  - etc.
- O projetista do workflow é quem decide qual(is) diretor(es) será(ão) usado(s)

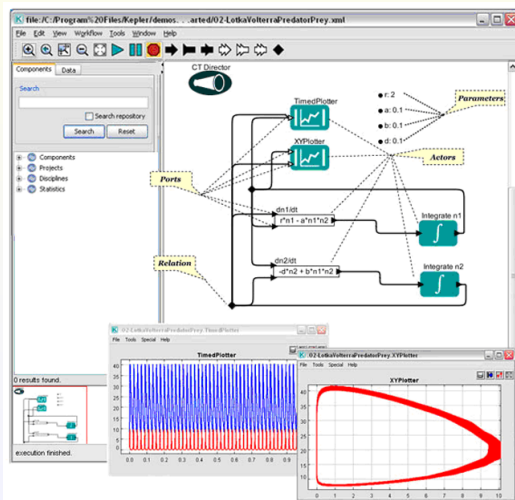
# Kepler

## Atores

- Especifica todos os tipos de ação que o workflow pode realizar
- Contêm *portas* que definem consumo ou produção de dados, interligadas por *canais*
- Algumas ações disponibilizadas por atores do Kepler:
  - acesso a serviços web (WebService),
  - leitura de dados do disco (FileFetcher, FileStager, etc.),
  - acesso a serviços de notificação (Email)
  - composição de vários atores (Composite)
  - etc.

# Kepler

## Interface gráfica



# Pegasus

- O Pegasus (*Planning for Execution in Grids*<sup>4</sup>) é um SGWC concebido para execução de workflows em plataformas distribuídas
- A partir da descrição do modelo do workflow e da informação dos recursos, o Pegasus faz o escalonamento, cria todas as tarefas necessárias para a implantação e execução do workflow
- O mesmo workflow pode ser executado em aglomerados, grades e plataformas de computação em nuvem sem modificações

---

<sup>4</sup>pegasus.

# Pegasus

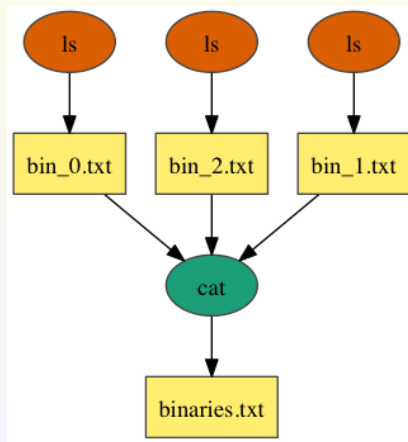
## Principais características

- Portabilidade** um mesmo workflow pode ser executado em diferentes plataformas
- Desempenho** o escalonador pode reordenar, agrupar e priorizar tarefas para melhorar o desempenho da execução
- Escalabilidade** o Pegasus já foi usado para executar workflows com milhões de tarefas
- Gerenciamento dos dados** o Pegasus é responsável por replicar, transferir e recuperar informações geradas pelas tarefas
- Confiabilidade** o Pegasus monitora a execução das tarefas e as reexecuta em caso de erros

# DAX

- A descrição do workflow é fornecida como um DAG descrito em XML, na linguagem DAX (*Directed Acyclic Graph in XML*)
- O DAX representa o fluxo de processamento e dependências entre as tarefas
- Cada atividade possui um identificador e metadados sobre seus arquivos de entrada e saída
- A linguagem DAX é puramente orientada a fluxo de dados

## Exemplo de workflow



## Exemplo do workflow em DAX

```

<adag>
  <job id="ID0000001" name="ls">
    <argument>l /bin</argument>
    <stdout name="bin_0.txt" link="output"/>
    <uses name="bin_0.txt" link="output"/>
  </job>
  <job id="ID0000002" name="ls">
    <argument>l /usr/bin</argument>
    <stdout name="bin_1.txt" link="output"/>
    <uses name="bin_1.txt" link="output"/>
  </job>
  <job id="ID0000003" name="ls">
    <argument>l /usr/local/bin</argument>
    <stdout name="bin_2.txt" link="output"/>
    <uses name="bin_2.txt" link="output"/>
  </job>
  <job id="ID0000004" name="cat">
    <argument>
      <file name="bin_0.txt"/>
      <file name="bin_1.txt"/>
      <file name="bin_2.txt"/>
    </argument>
    <stdout name="binaries.txt" link="output"/>
    <uses name="bin_2.txt" link="input"/>
    <uses name="bin_0.txt" link="input"/>
    <uses name="binaries.txt" link="output" register="false" transfer="true"/>
    <uses name="bin_1.txt" link="input"/>
  </job>
</adag>

```

# Alguns grupos que pesquisam sobre Workflows Científicos

## No Brasil

- Universidade Federal do Rio de Janeiro (PESC-COPPE-UFRJ), em parceria com o Laboratório Nacional de Computação Científica (LNCC)
  - Criadores das ferramentas Chiron e SciCumulus
- Universidade Estadual de Campinas (IC-UNICAMP)
- Universidade de São Paulo (DCC-IME-USP)

# Alguns grupos que pesquisam sobre Workflows Científicos

## Fora do Brasil

- University of Southern California (Information Sciences Institute)
  - Profa. Ewa Deelman
  - Criadores do Pegasus
- University of Manchester (School of Computer Science)
  - Profa. Carole Goble
  - Criadores do Taverna, myGrid, myExperiment
- University of Chicago (Computation Institute)
  - Prof. Ian Foster
  - Criadores do conceito de grade e do projeto Globus
- University of Innsbruck (Institute of Computer Science)
  - Prof. Thomas Fahringer
  - Criadores do ASKALON

Obrigada!

**Kelly R. Braghetto**

`kellyrb@ime.usp.br`

`www.ime.usp.br/~kellyrb`