

# Taverna , reloaded

Paolo Missier, Stian Soiland-Reyes, Stuart Owen, Alex Nenadic,  
Ian Dunlop, Alan Williams, Carole Goble

School of Computer Science  
University of Manchester, UK

Wei Tan

Argonne National Laboratory  
Argonne, IL, USA

Tom Oinn  
EBI, UK

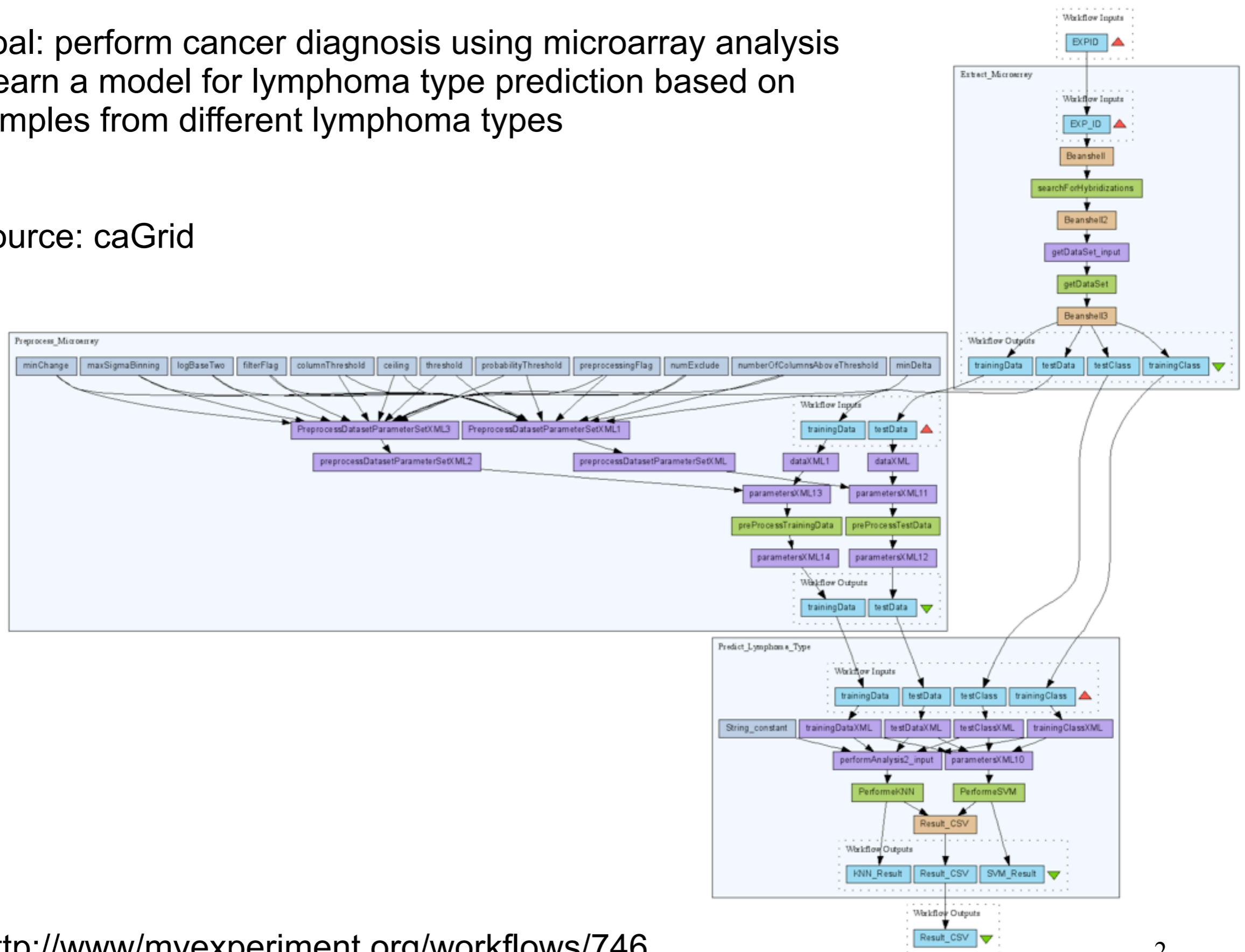
---

SSDBM

Heidelberg, Germany  
June 30 - July 2, 2010

Goal: perform cancer diagnosis using microarray analysis  
 - learn a model for lymphoma type prediction based on samples from different lymphoma types

source: caGrid



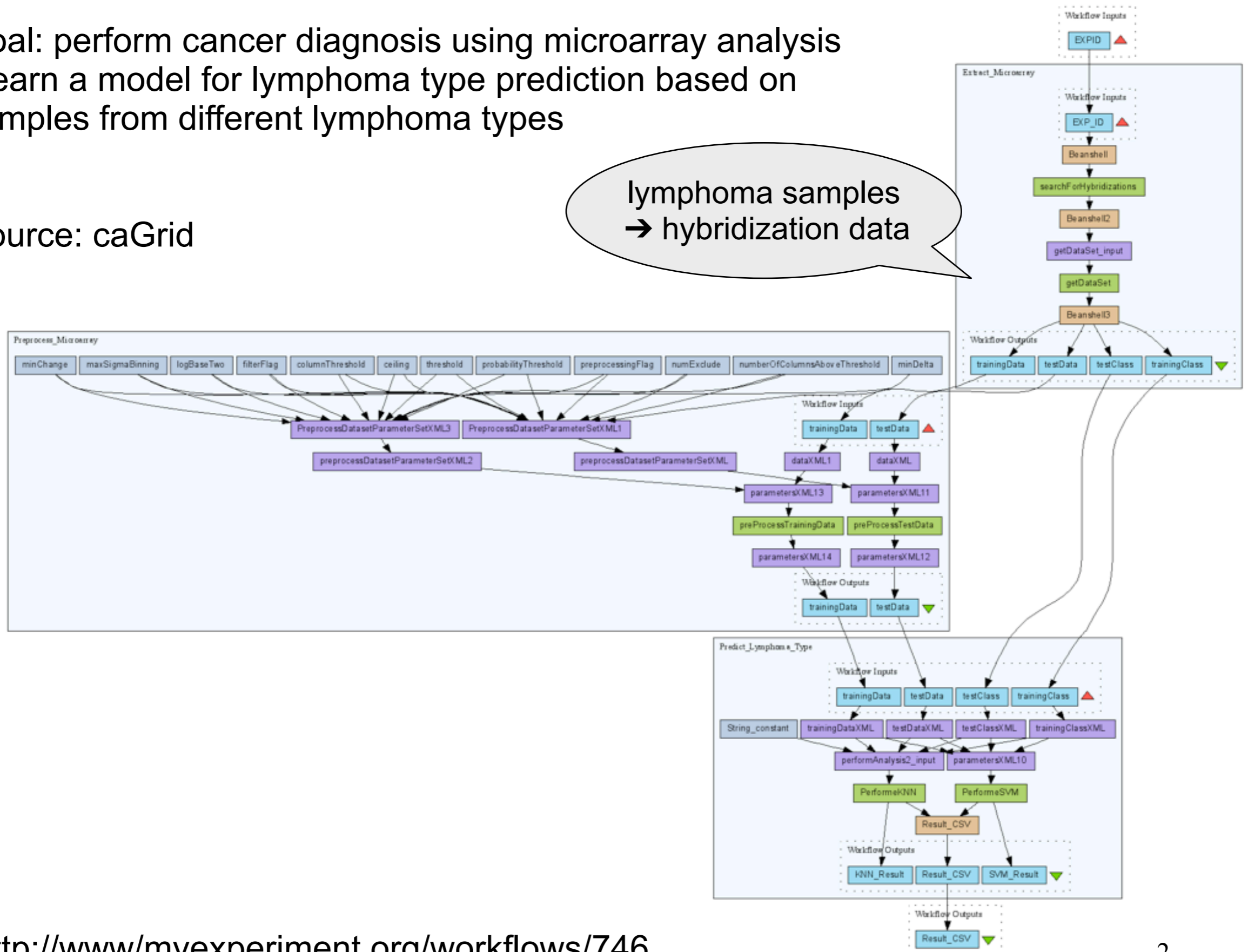
<http://www.myexperiment.org/workflows/746>

# Taverna: scientific workflow management

Goal: perform cancer diagnosis using microarray analysis  
 - learn a model for lymphoma type prediction based on samples from different lymphoma types

source: caGrid

lymphoma samples  
 → hybridization data

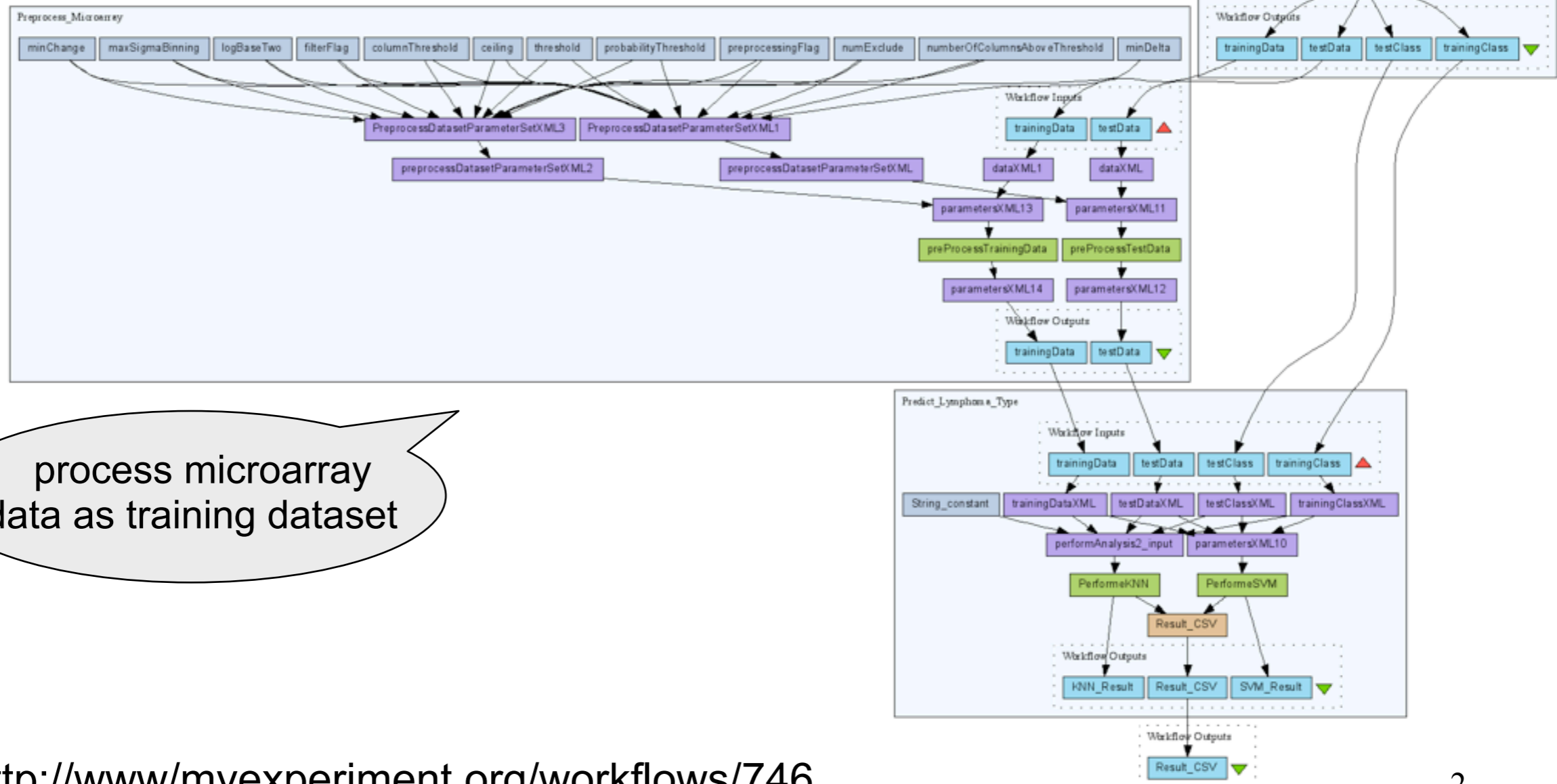


# Taverna: scientific workflow management

Goal: perform cancer diagnosis using microarray analysis  
 - learn a model for lymphoma type prediction based on samples from different lymphoma types

source: caGrid

lymphoma samples  
 → hybridization data



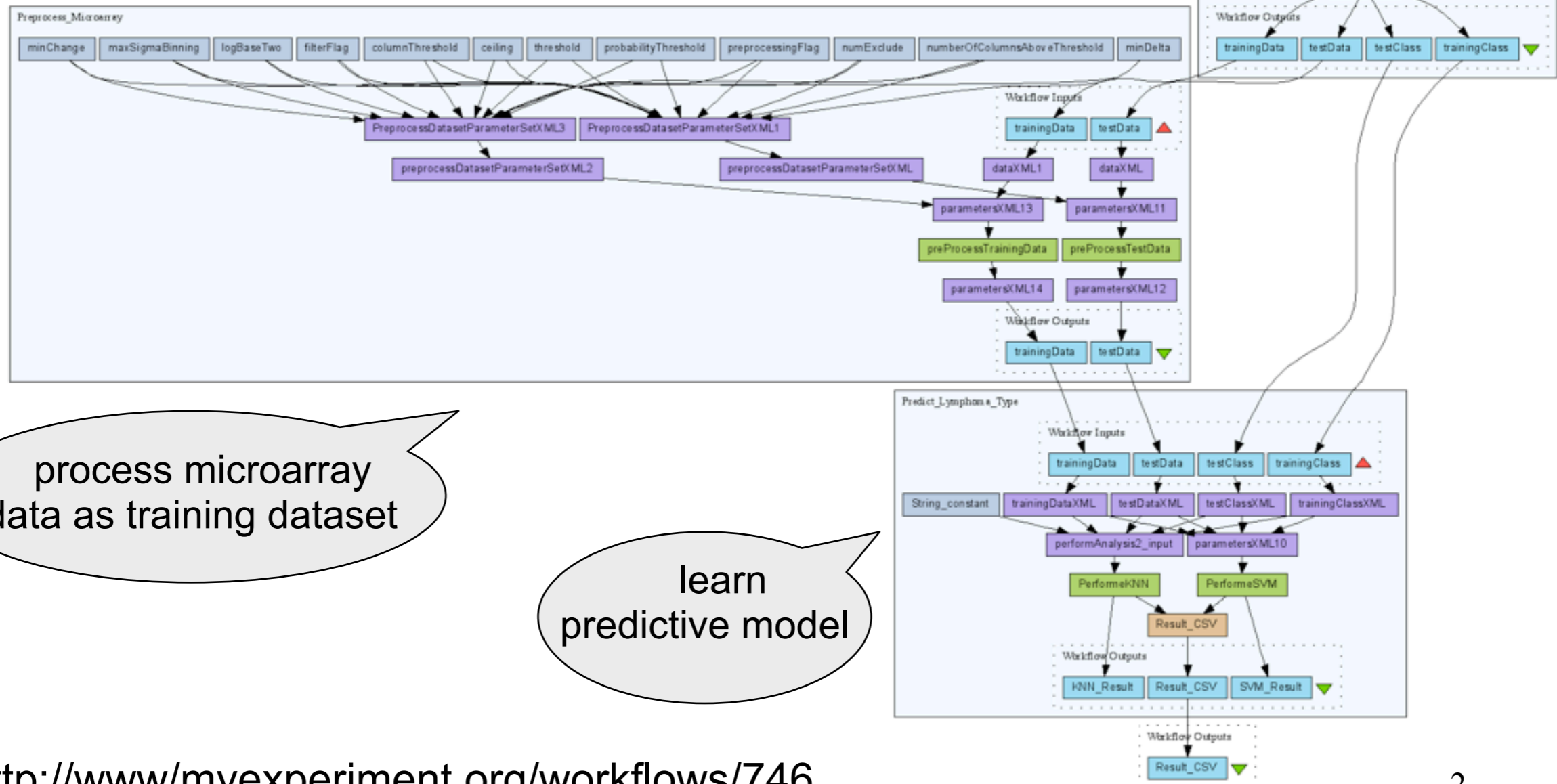
process microarray data as training dataset

# Taverna: scientific workflow management

Goal: perform cancer diagnosis using microarray analysis  
 - learn a model for lymphoma type prediction based on samples from different lymphoma types

source: caGrid

lymphoma samples  
 → hybridization data

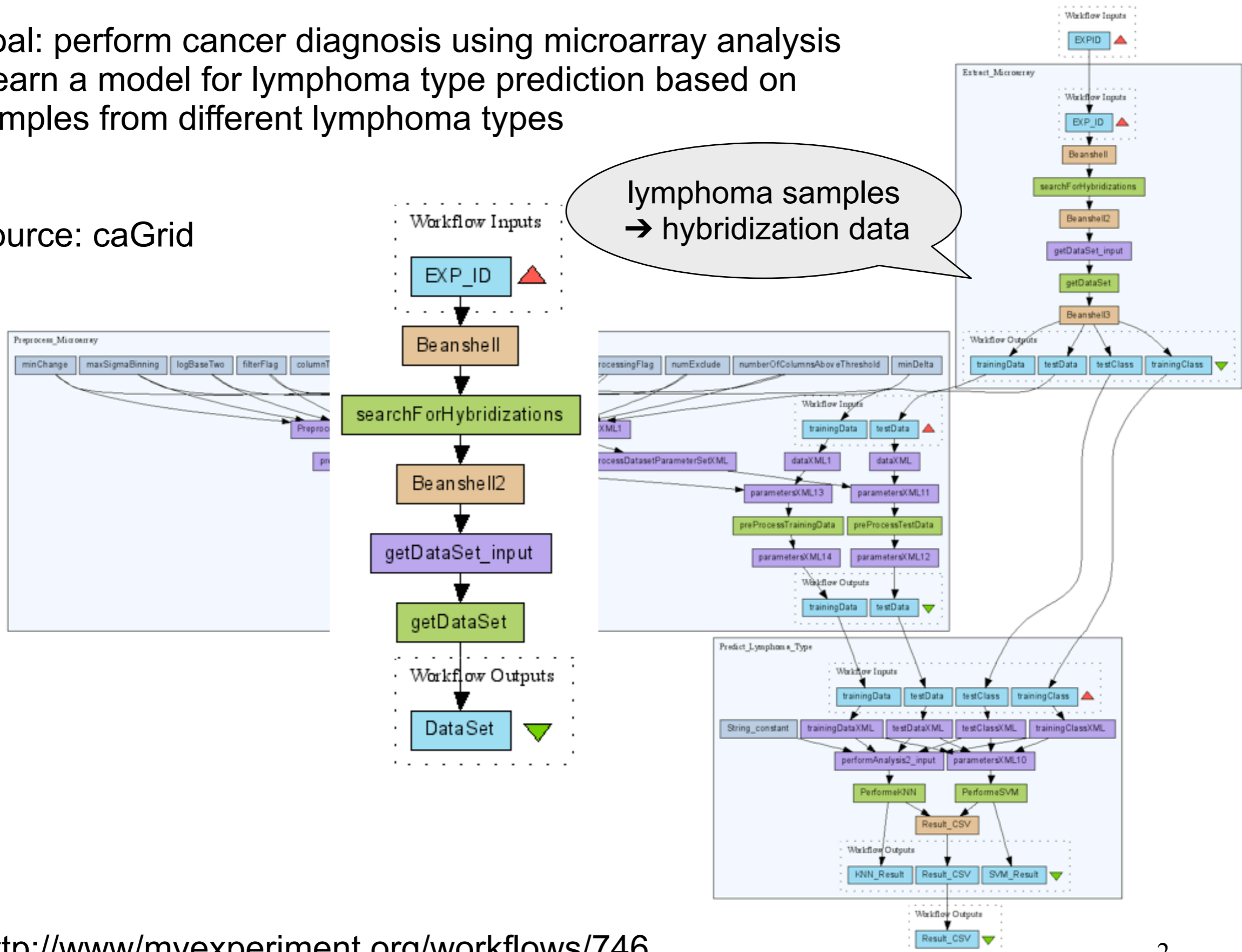


# Taverna: scientific workflow management

Goal: perform cancer diagnosis using microarray analysis  
 - learn a model for lymphoma type prediction based on samples from different lymphoma types

source: caGrid

lymphoma samples  
 → hybridization data

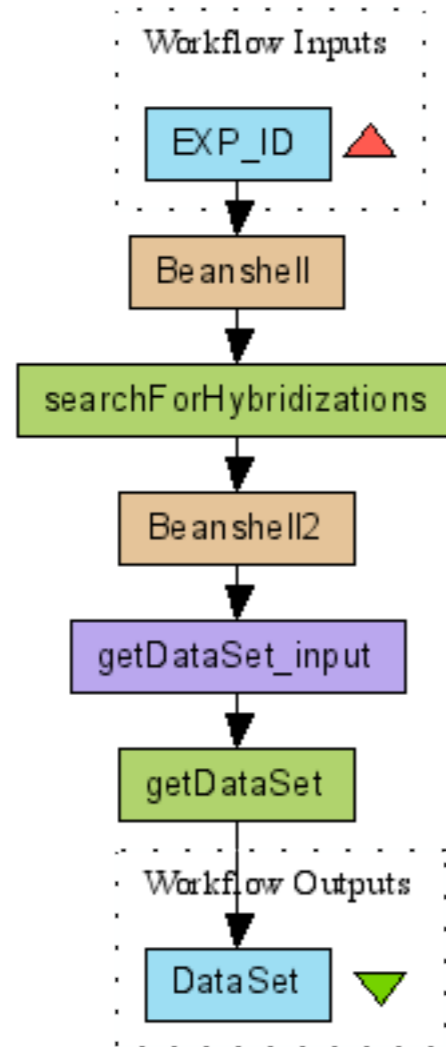


- Taverna for data-intensive science
  - architectural goals... and how they are being achieved
- Performance figures on benchmark workflows

- **Scalability**
  - size of input data / size of input collections
- **Configurability**
  - each processor in the workflow individually tunable to adjust to the underlying platform
- **Extensibility**
  - plugin architecture to accommodate specific service groups
    - caGrid, BioMoby
  - extensible set of operators that define the semantics of the model

# Opportunities for parallel processing

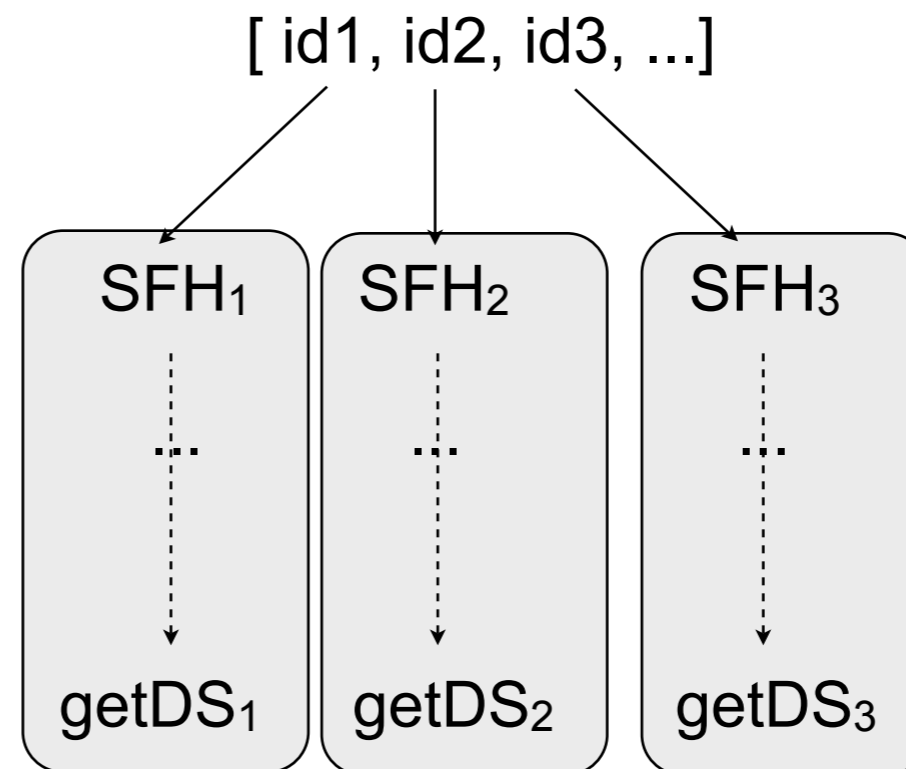
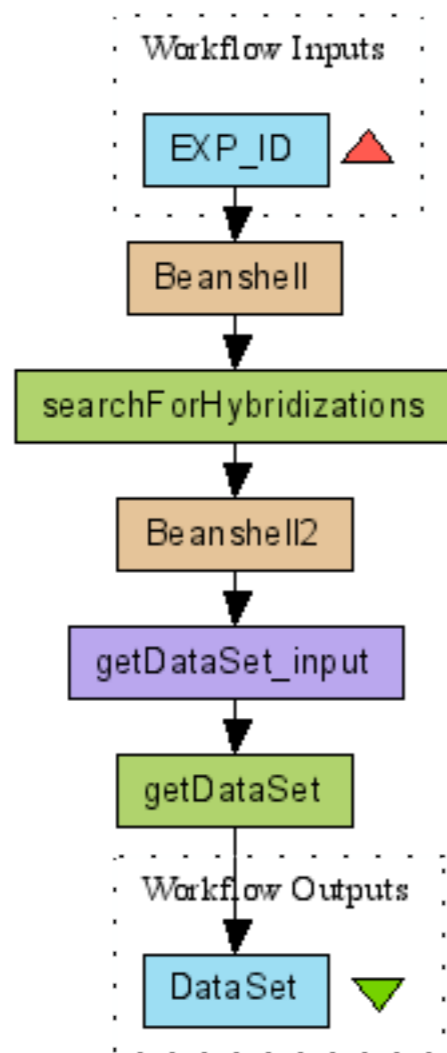
- intra-processor: implicit iteration over list data
- inter-processor: pipelining



[ id1, id2, id3, ...]

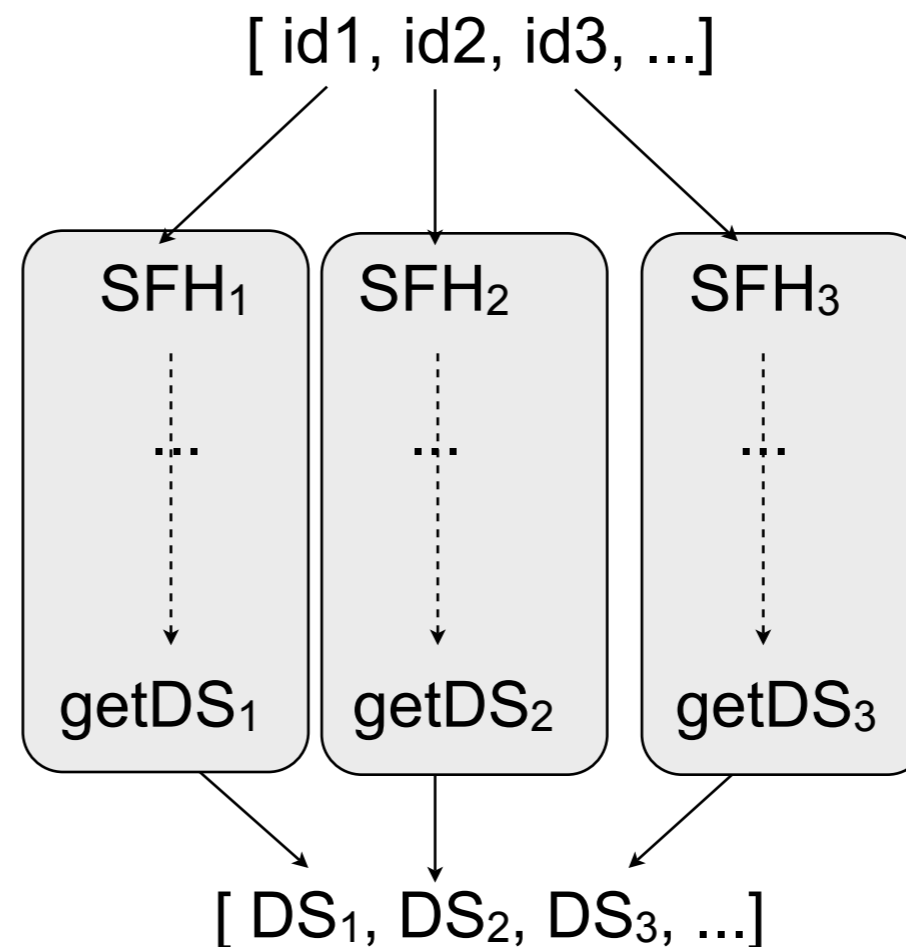
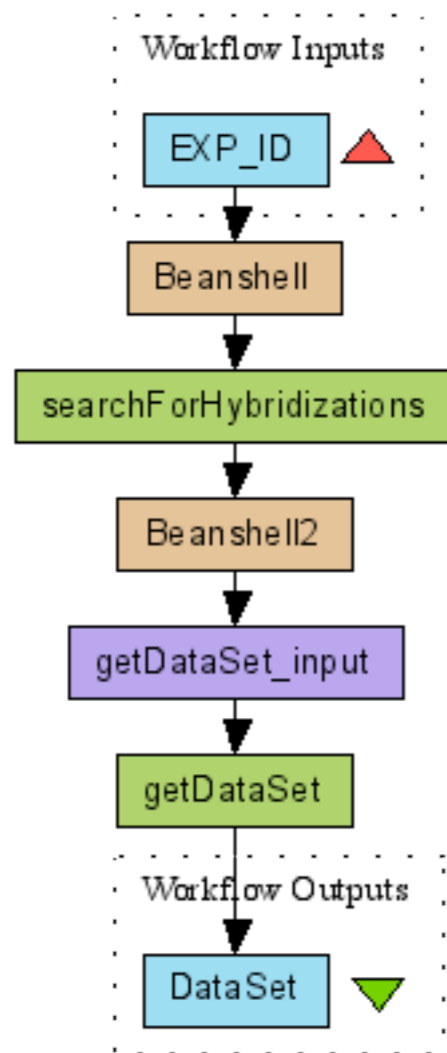
# Opportunities for parallel processing

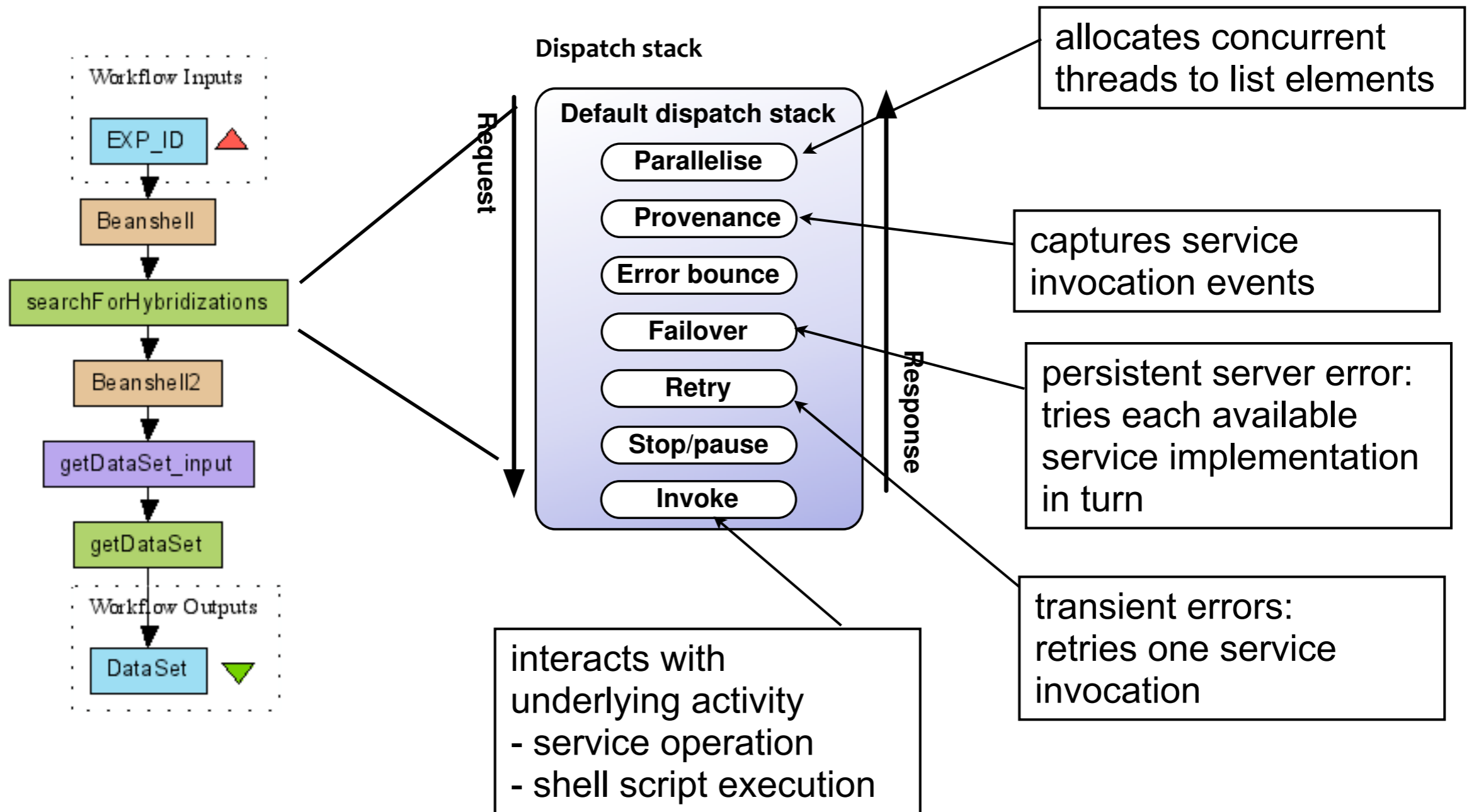
- intra-processor: implicit iteration over list data
- inter-processor: pipelining

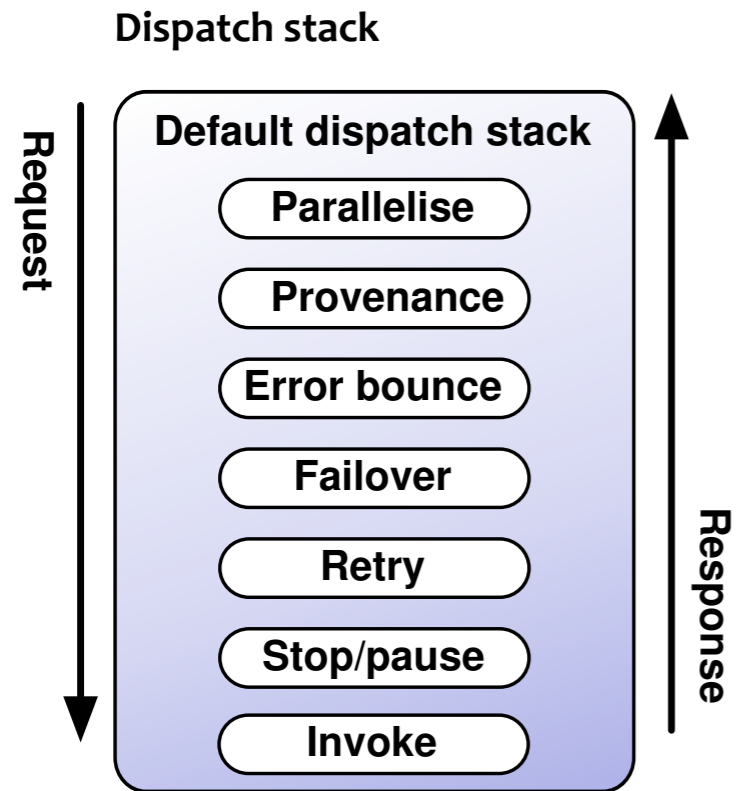


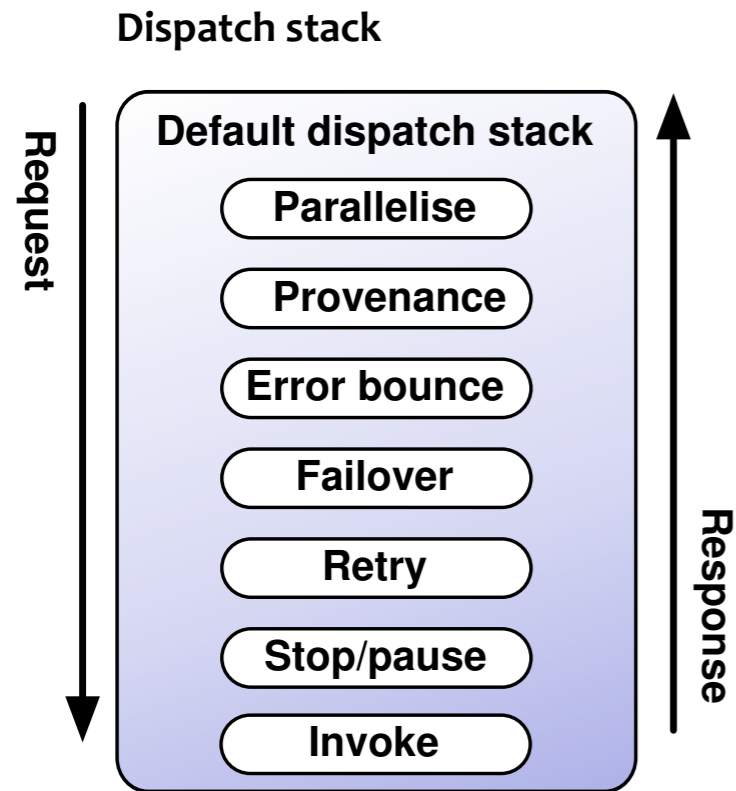
# Opportunities for parallel processing

- intra-processor: implicit iteration over list data
- inter-processor: pipelining

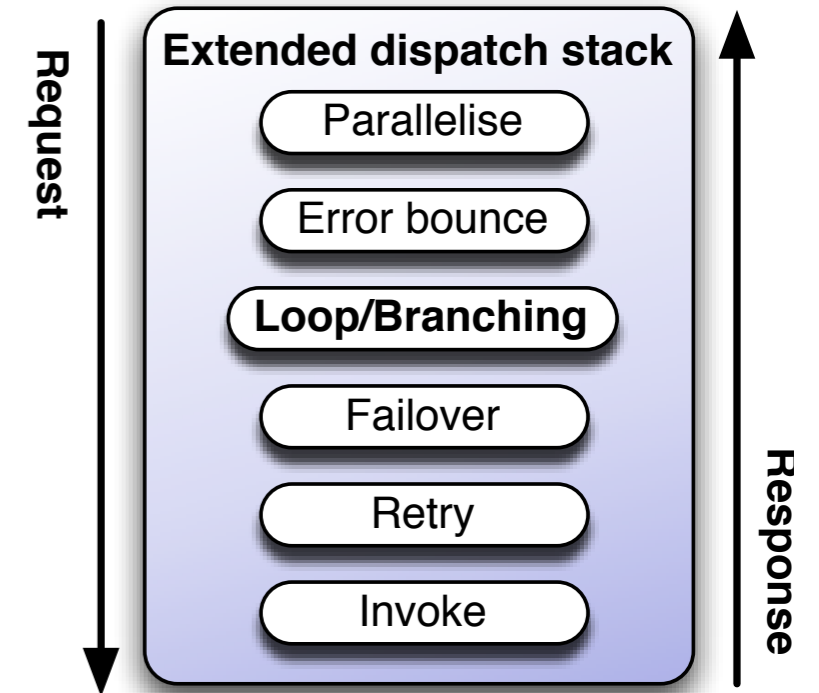
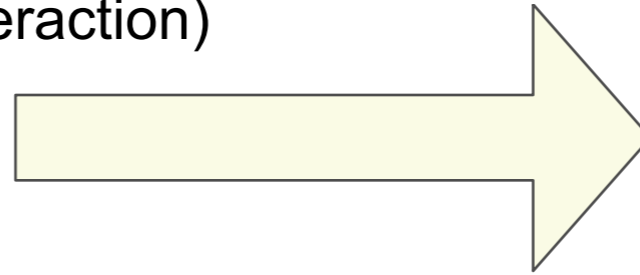




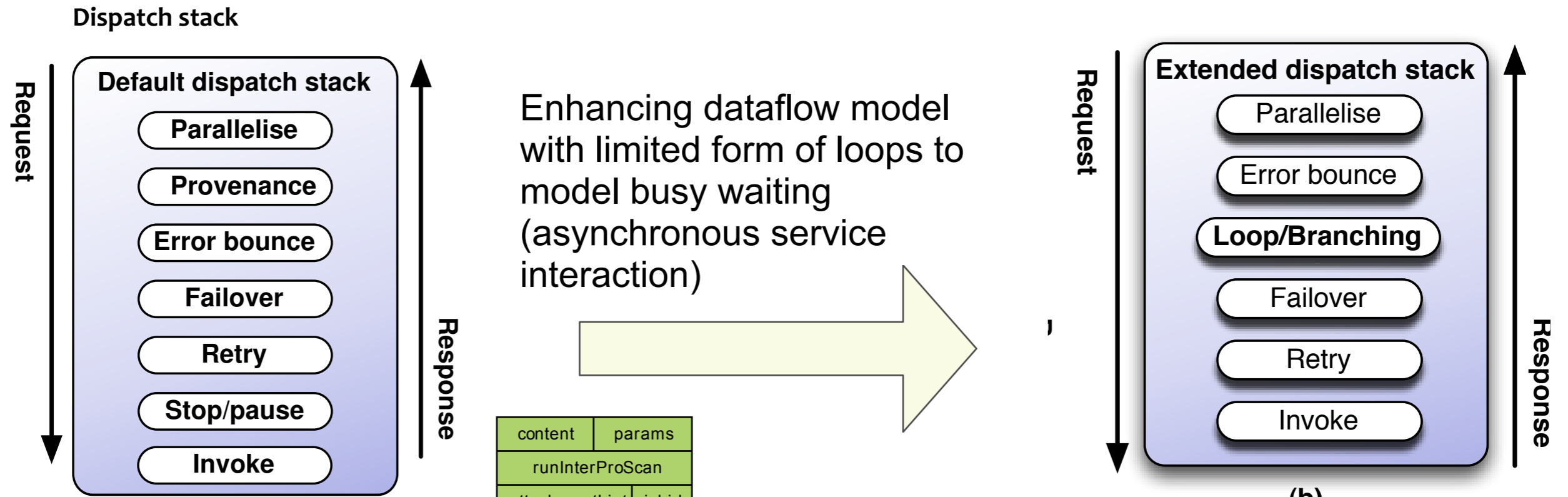




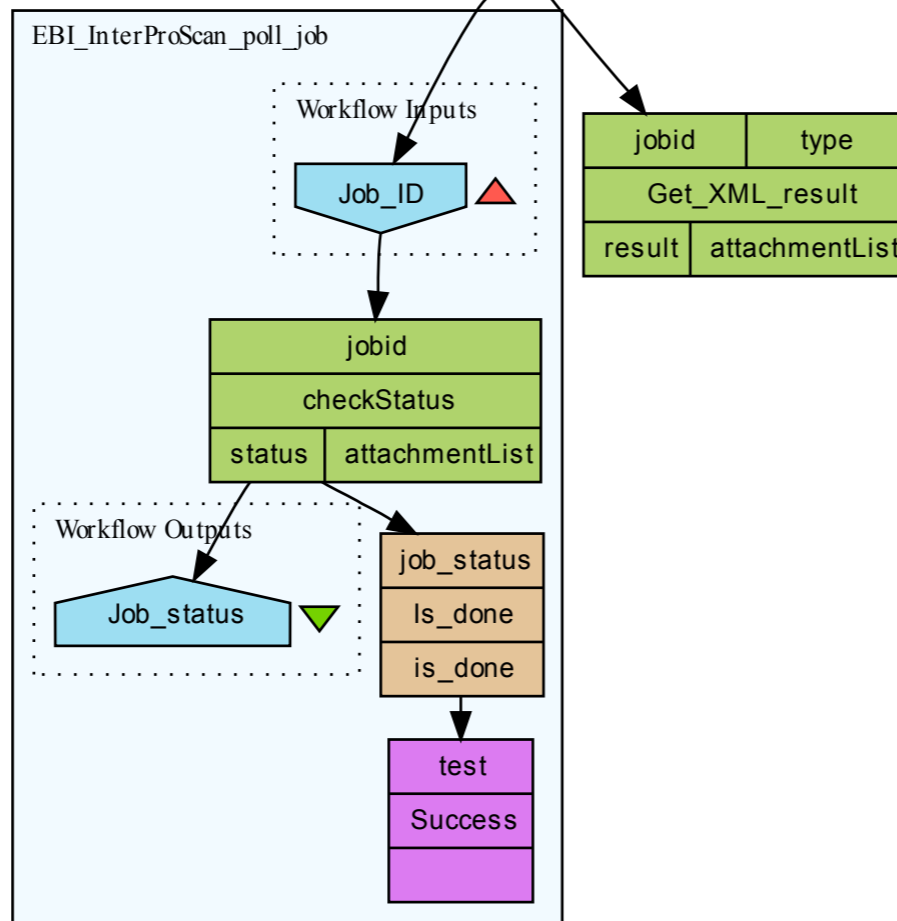
Enhancing dataflow model  
with limited form of loops to  
model busy waiting  
(asynchronous service  
interaction)

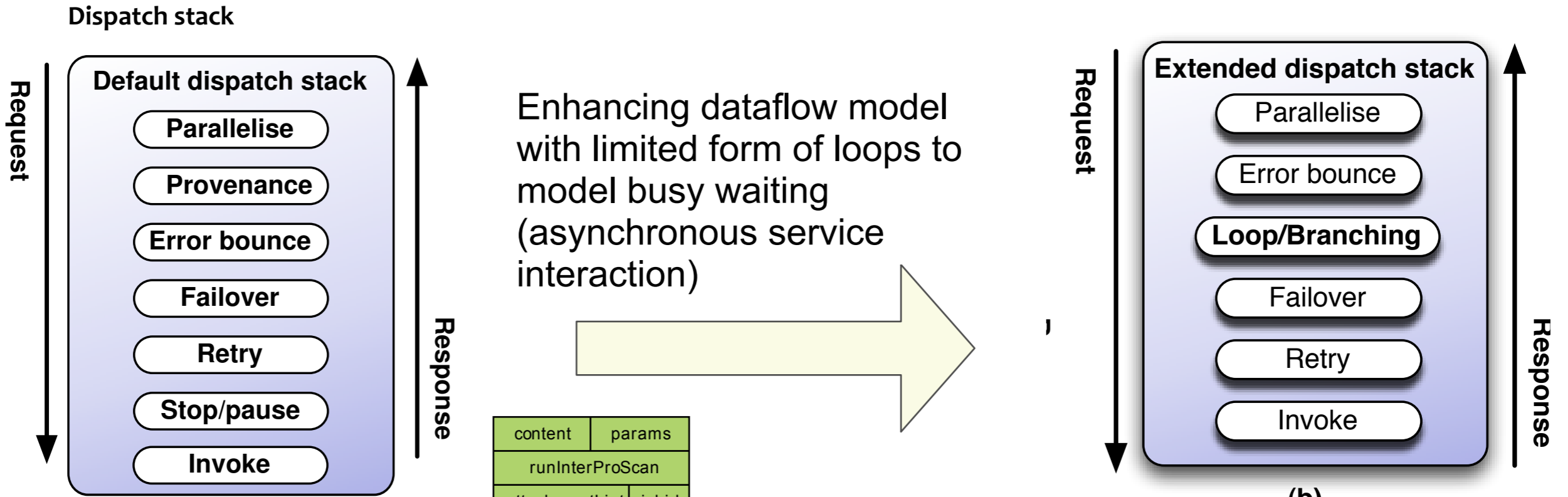


(b)

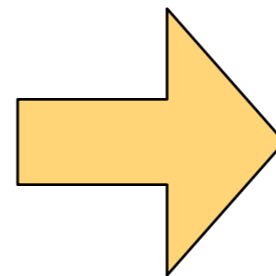
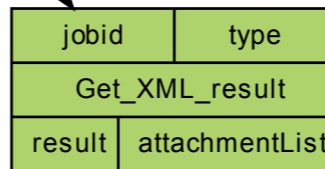
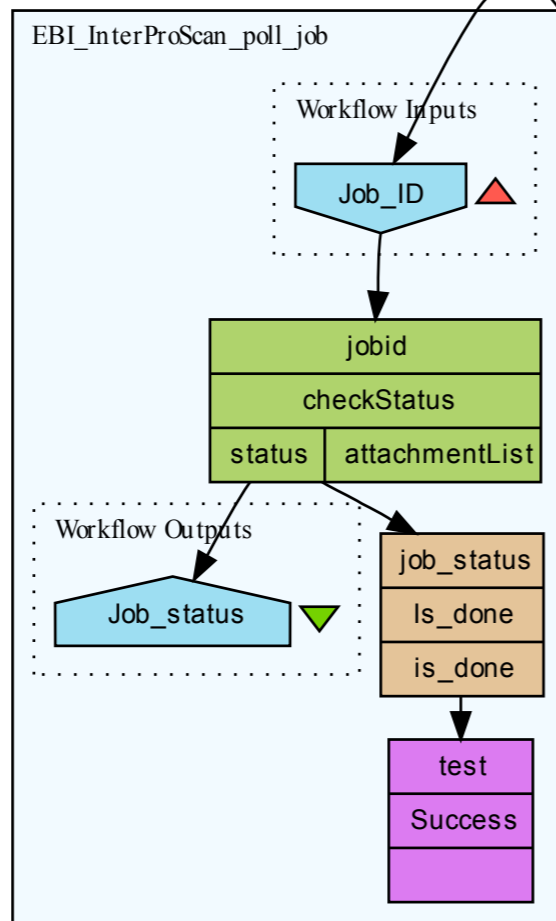


Busy waiting using an explicit workflow pattern

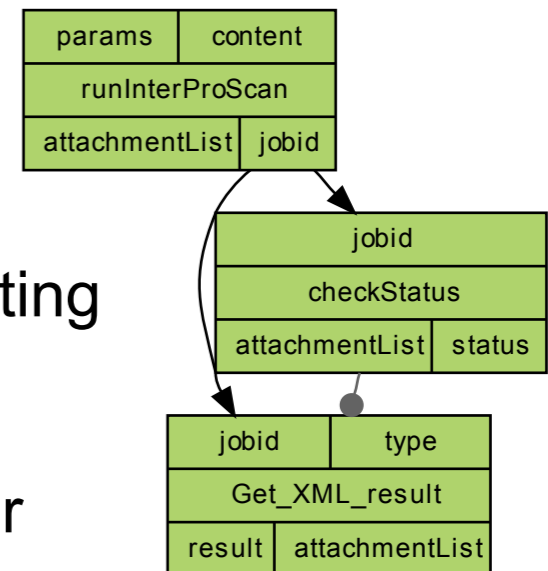




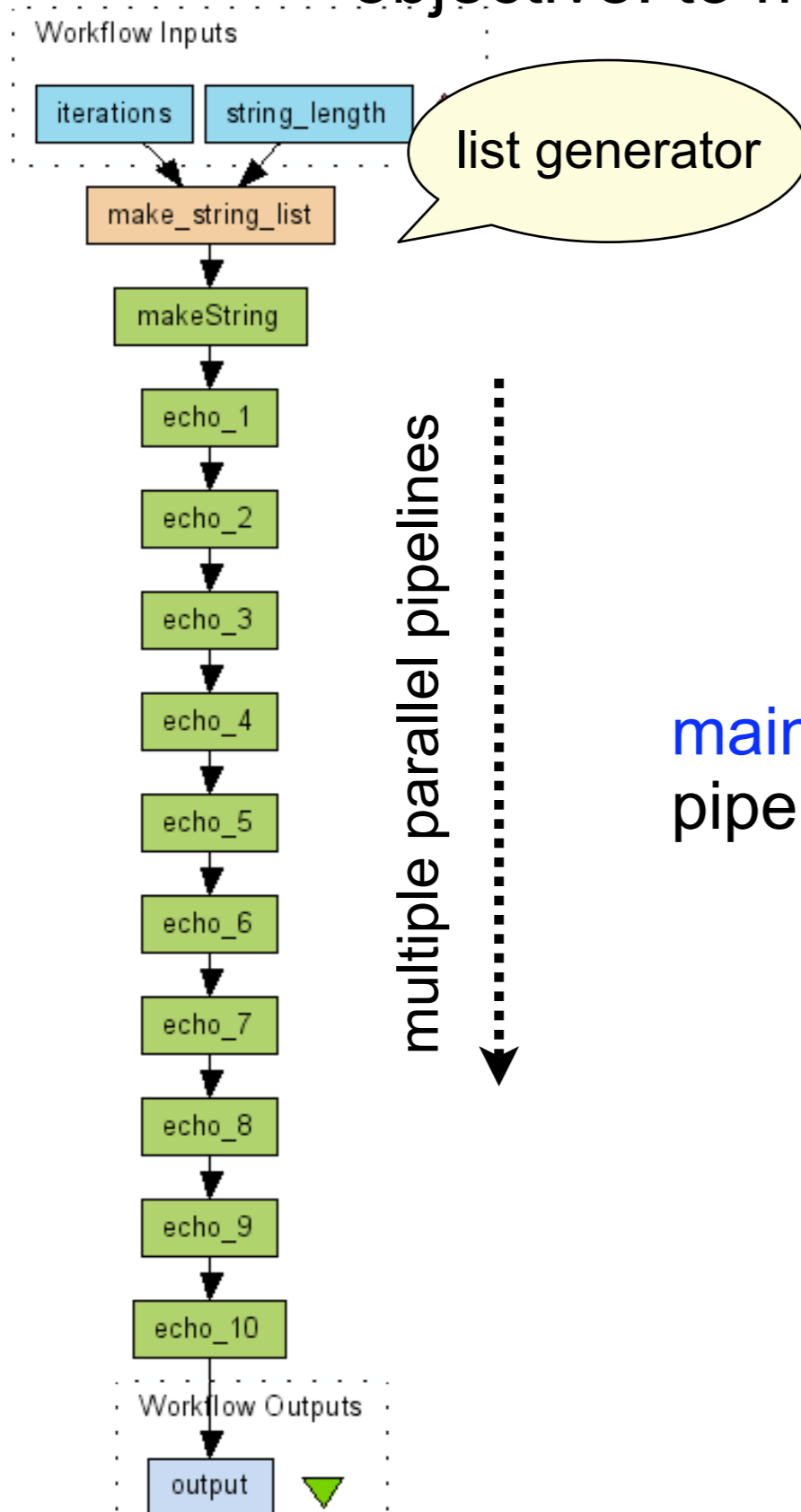
Busy waiting using an explicit workflow pattern



Busy waiting using an loop processor



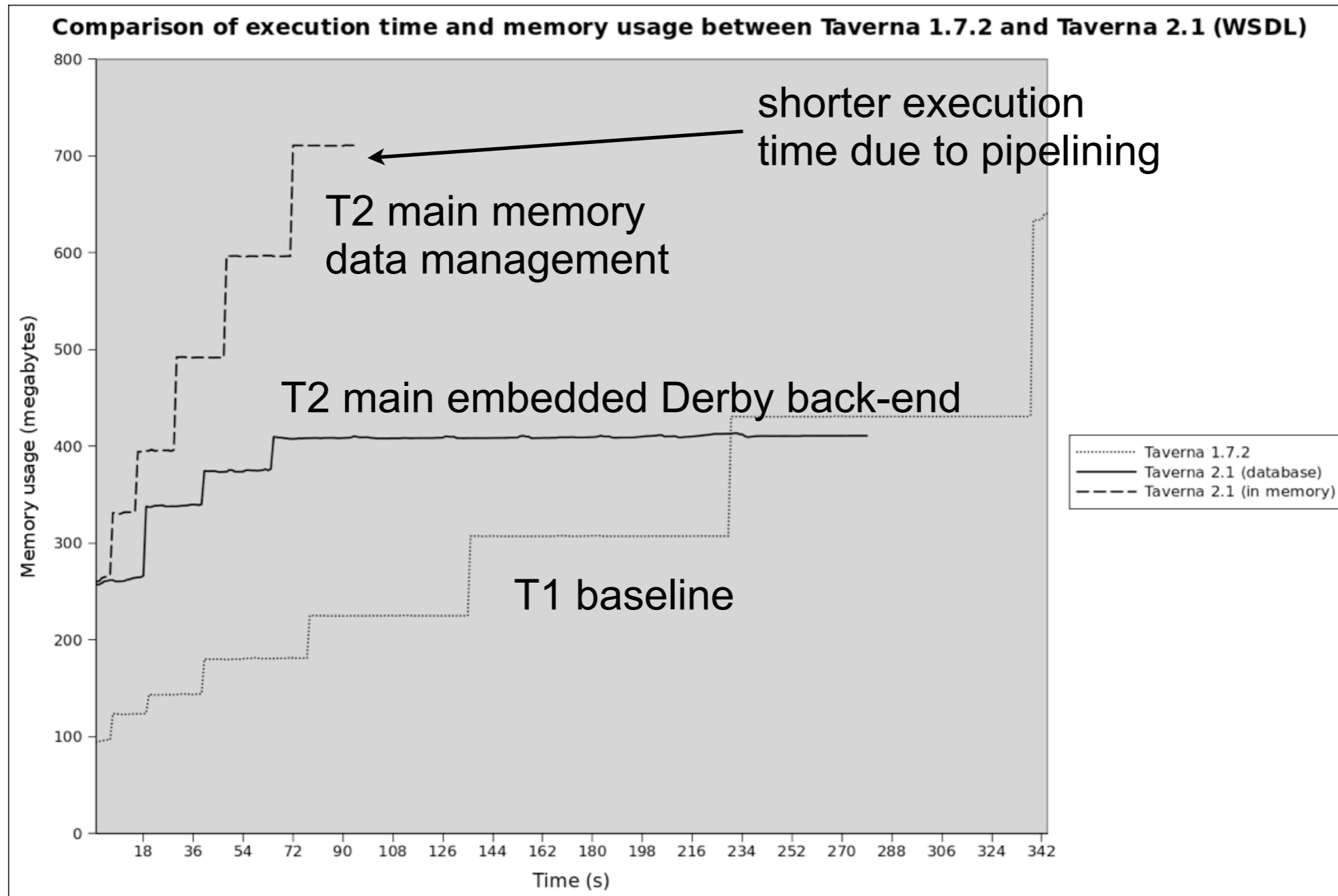
- previous version of Taverna engine used as baseline
- objective: to measure incremental improvement



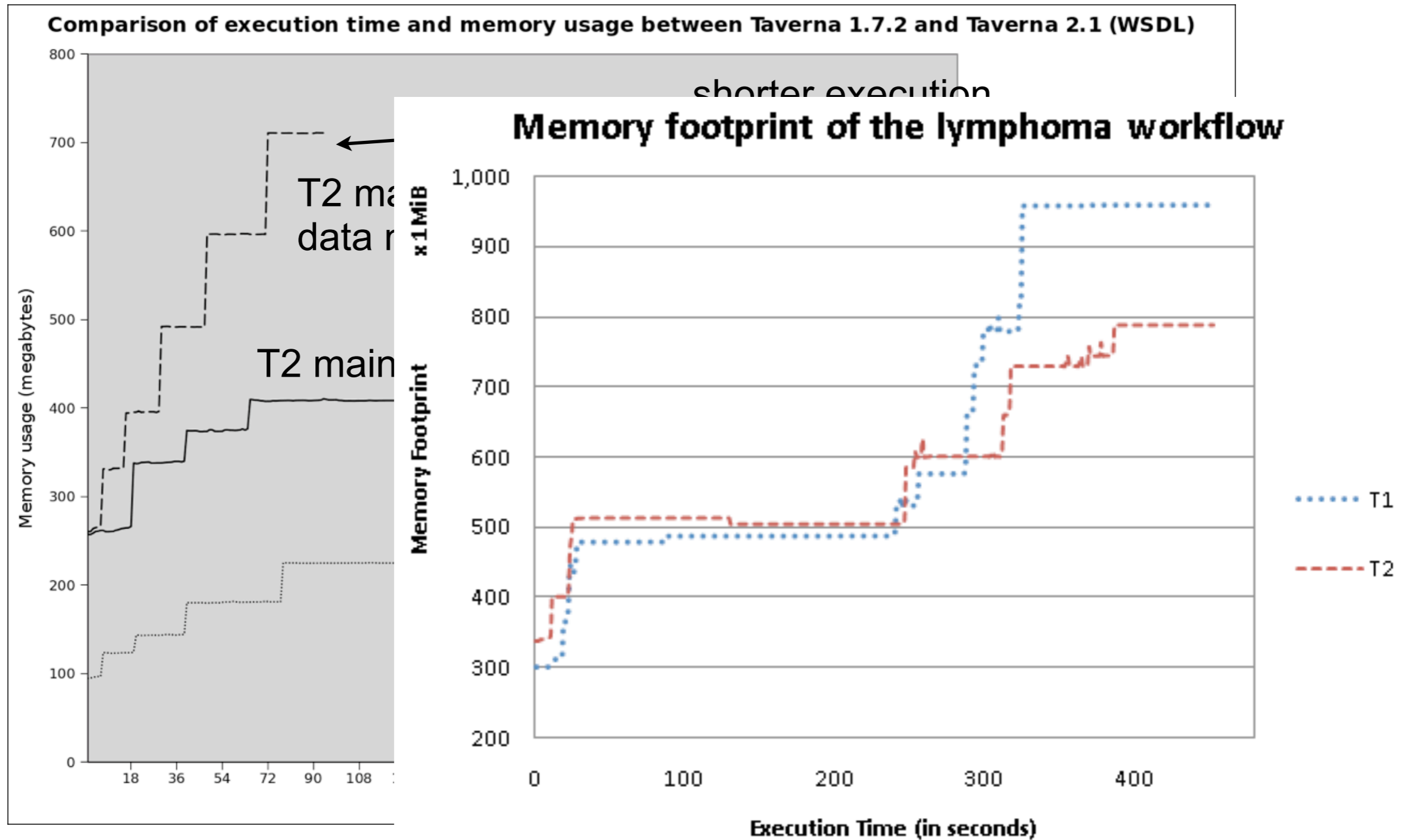
Parameters:

- byte size of list elements (strings)
- size of input list
- length of linear chain

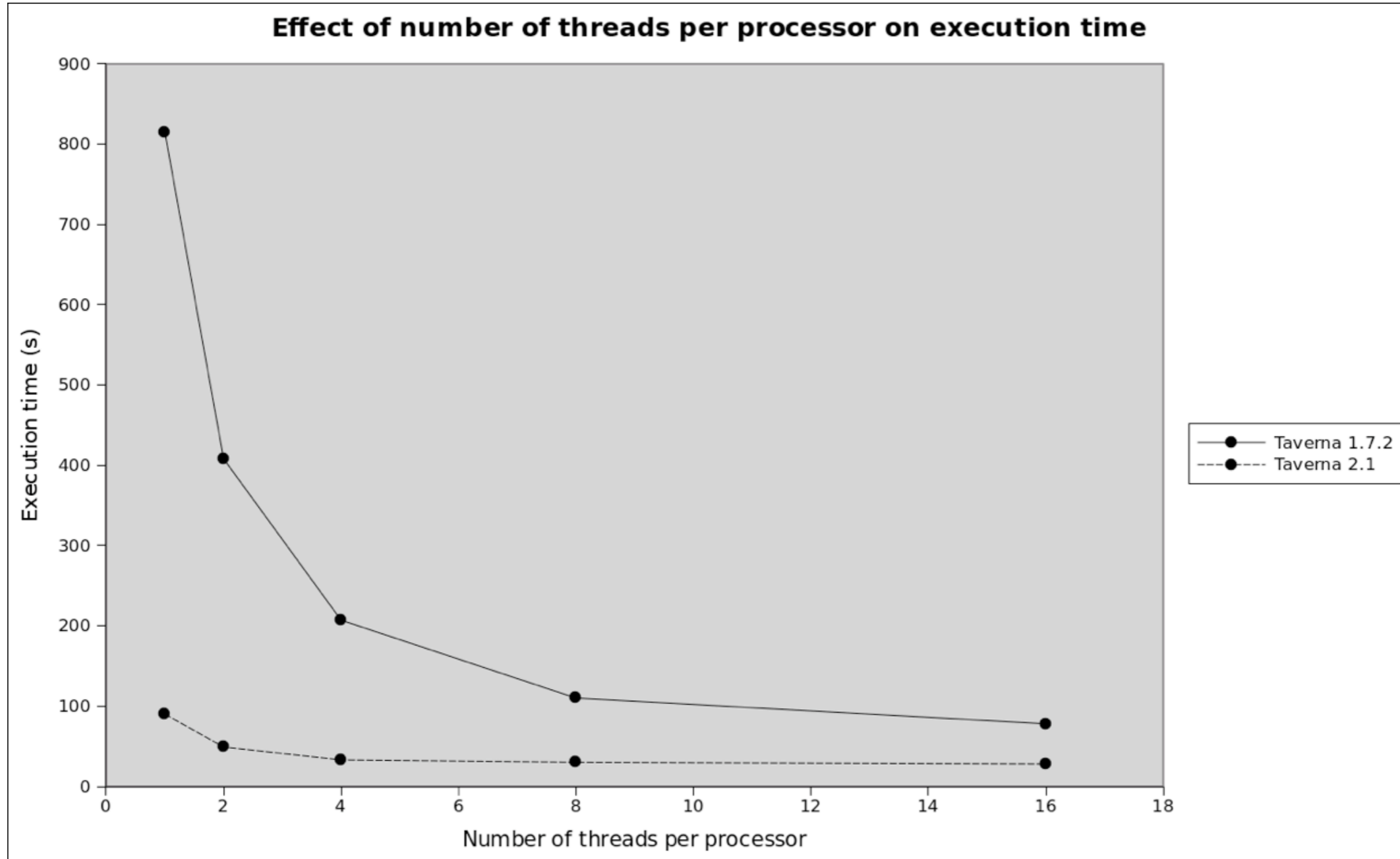
**main insight:** when the workflow is designed for pipelining, parallelism is exploited effectively



list size: 1,000 strings of 10K chars each  
no intra-processor parallelism (1 thread/processor)

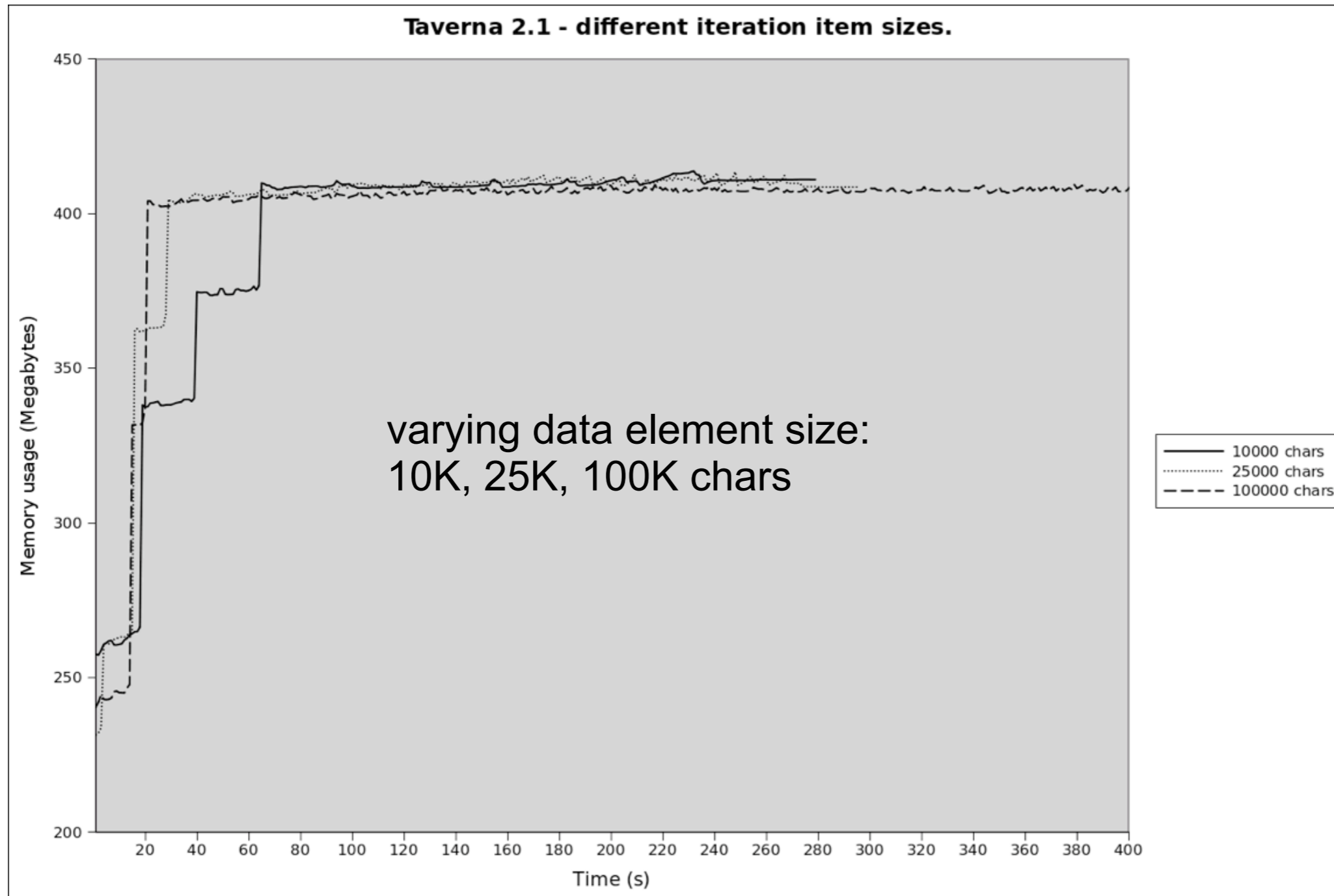


list size: 1,000 strings of 10K chars each  
no intra-processor parallelism (1 thread/processor)



pipelining in T2 makes up for smaller pools of threads/processor

Separation of data and process spaces ensures scalable data management



- Taverna 2 re-engineered for scalability on data-intensive pipelined dataflows
- separation of data and process spaces
- generic stack pattern for principled extensions
  - limited while loop, if-then-else
  - provenance capture
  - ... open plugin architecture accommodates further extensions

For further details and a nice chat:  
please visit our poster!!