SPAN:
System for Phyloinformatics ANalysis using Ontologies and Automated Reasoning

Arlin Stoltzfus
Gopal Gupta
Enrico Pontelli

PART I: OVERVIEW

Motivations and Project Objectives

- Goals
  - Develop a comprehensive infrastructure for evolutionary informatics
  - Enable more widespread use of evolutionary comparative analysis methods in functional inferences
- Problem identification
  - Multitude of data representation formats
    - concentrated on individual aspects (e.g., nucleotide sequence, alignment)
    - legacy textual formats
      - complex to process (NEXUS – context sensitive grammar)
      - poorly described (PAML complex output)
      - poorly advertised (CHADO)
    - Yet, types of data are very similar (alignments, node-wise attributes, column-wise attributes, transition models…)
  - Multitude of applications
    - task-specific
    - complex and non-standard interfaces (e.g., PAML)
    - Limited ability to develop workflows
      - hand-shepherd data
      - “closed” packaged frameworks (e.g., MEGA)
      - explicit low-level coding (BioPerl)
    - Yet, generalized systems are possible (HyPhy, Taverna)
  - Validation and quality control
    - lack of methods to assess quality of analyses (e.g., no reference standards)
    - reproducibility of analyses
  - Delayed absorption of technological advancement in computing

NIST UTD NM STATE UNIVERSITY

SPAN

- An *integrated* and *open* architecture to address (most of) these issues
- at this time, mostly a wish list…
- lead participants:
  - Arlin Stoltzfus
  - Gopal Gupta
  - Enrico Pontelli

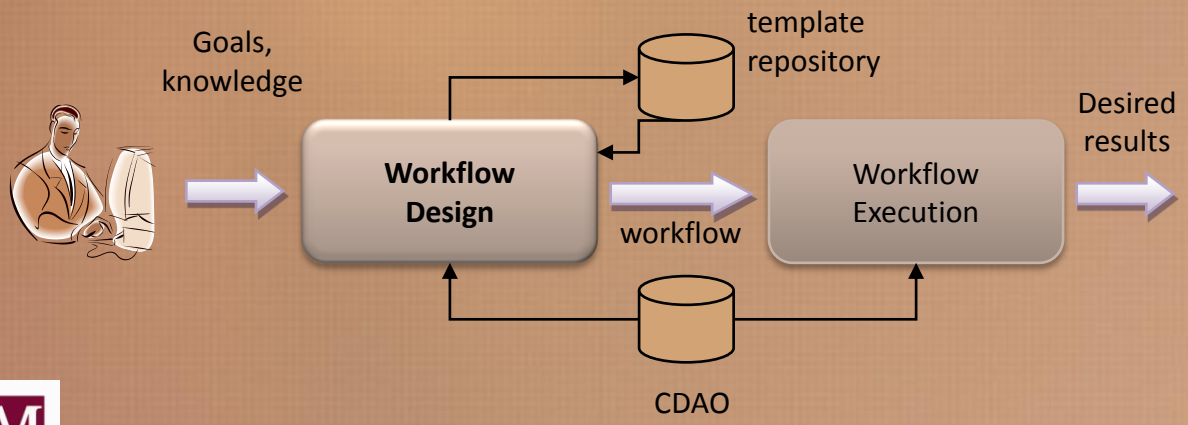
SPAN: Guiding Principles and Proposed Technology

- Design Principles
  - Inter-operation
    - Data-level Inter-operation
    - Operation-level Inter-operation
  - Workflows
    - High-level design
    - Reusability
  - Use-case-driven Development
- Proposed Technology
  - Ontology
    - as inter-operation artifact
    - as a “language” for describing analysis processes
  - Domain Specific Languages
    - from ontology to workflows
  - Workflow Infrastructure
    - Web Services
    - Automated Reasoning
    - BioPerl

NIST UTD NM STATE UNIVERSITY

SPAN: the “big picture”

- Workflow design infrastructure (domain-specific features)
- Workflow execution infrastructure
- Comparative Data Analysis Ontology (CDAO) as the underlying “language”



Comparative Data Analysis Ontology (CDAO)

- CDAO
  - capture concepts and relations that describe relevant data entities in evolutionary analysis (*structure ontology*)
  - capture concepts and relations describing relevant transformations in evolutionary analysis (*process ontology*)
- Why?
  - formalize concepts and relations
  - creates a common language (e.g., facilitates inter-operation, guides new developments)
- Design and implementation approaches
  - Design Strategy:
    1. study use cases (e.g., NESCent Phyloinformatics hackaton)
    2. identify core concepts and relations (data and operations)
    3. identify related ontologies (e.g., MAO)
    4. formalize ontology and implement its representation (e.g., OWL)
    5. interface ontology to legacy data formats
    6. iterative evaluation and challenge on demonstration projects
  - Implementation Technology
    - OWL and related tools for representation
    - Logic Programming as a reasoning mechanism
    - Logic Programming to create inter-operation tools
  - Some requirements
    - provide a concept coverage adequate to capture descriptions of established data formats (NEXUS, MEGA, Phylip, …)
    - integrate with OBO and follow OBO guidelines (e.g., OBO_REL)
    - provide easy to use APIs to converse between the ontology and existing documents

NIST UTD NM STATE UNIVERSITY

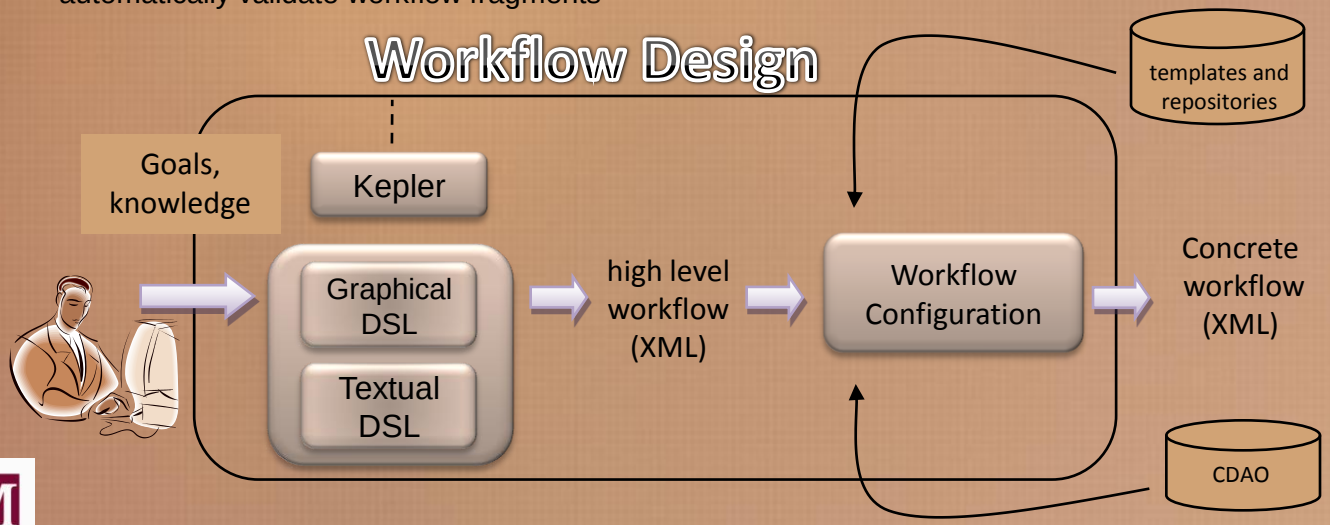
Workflows

- Workflows as sequences of transformations
  - all inter-operability problems have been resolved
- Growing literature of workflow development infrastructure for bioinformatics
  - Kepler
  - BioMoby
  - Taverna
  - …
- Design principles
  - Workflow design environment
    - Domain specific – rely on CDAO concepts
    - Compatible with existing workflow environments (e.g., Kepler)
    - Capable of operating at different levels of abstraction
    - Textual and graphical
  - templates
    - libraries of workflows and workflow templates
  - automated configuration
  - execution
    - web services and composition

NIST UTD NM STATE UNIVERSITY

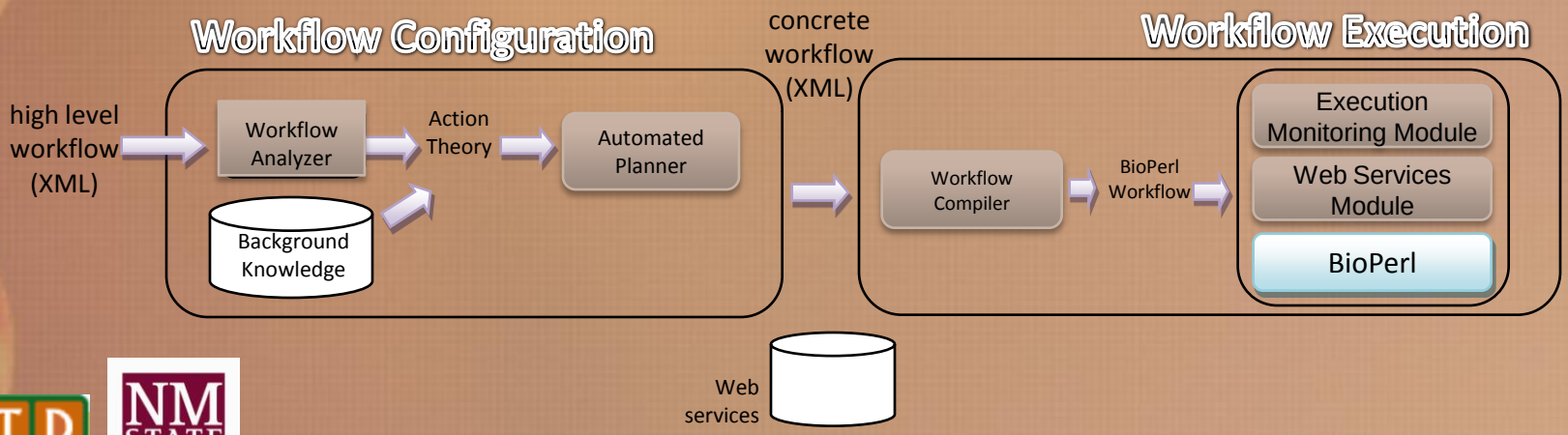
Workflows: Design

- three modalities
  - *templated*: repositories of workflows and workflow templates
    - searchable
    - customizable (e.g., I/O)
    - possibly imported from other frameworks
  - *designed*: create a new workflow or modify an existing template
    - select operations and control structures from CDAO
    - manually compose them into a workflow
  - *discovered*: automated reasoner assists in the construction of a workflow
    - searches service repositories based on user queries
    - automatically inserts glue code
    - automatically suggests compatible services
    - automatically validate workflow fragments



Workflows: Configuration and Execution

- Configuration: process of enabling creation of a concretely executable workflow
  - selection of services for the various operations
  - composition
  - evaluation w.r.t. quality criteria and user models
- Execution:
  - workflow compilation to enhanced BioPerl



Feasibility Test Projects

- Preliminary projects to demonstrate feasibility of proposed technology
  - Ontology-mediated formats interoperation
    - Multiple sequence alignments
    - MAO
    - NEXUS, Phylip, MEGA APIs http://www.cs.nmsu.edu/~bchisham/Ontology/src/web/ontology.php
  - Ontology design
    - in progress…
  - Semantic-based Web service description and composition
  - PhyLOG
    - web service infrastructure to execute bioinformatics workflows

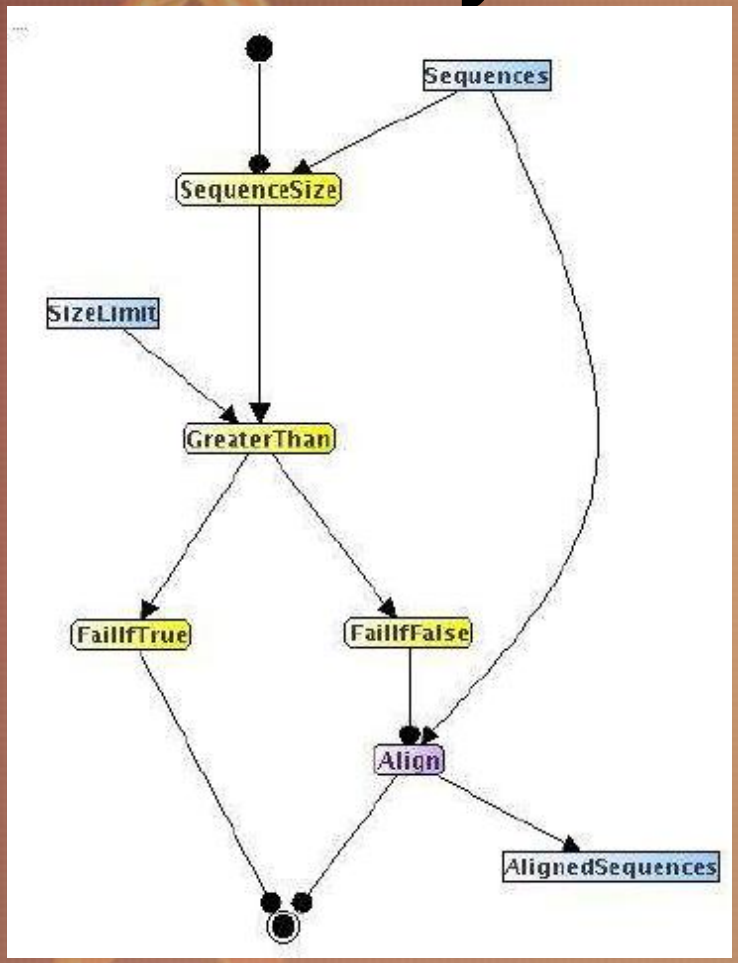
NIST UTD NM STATE UNIVERSITY

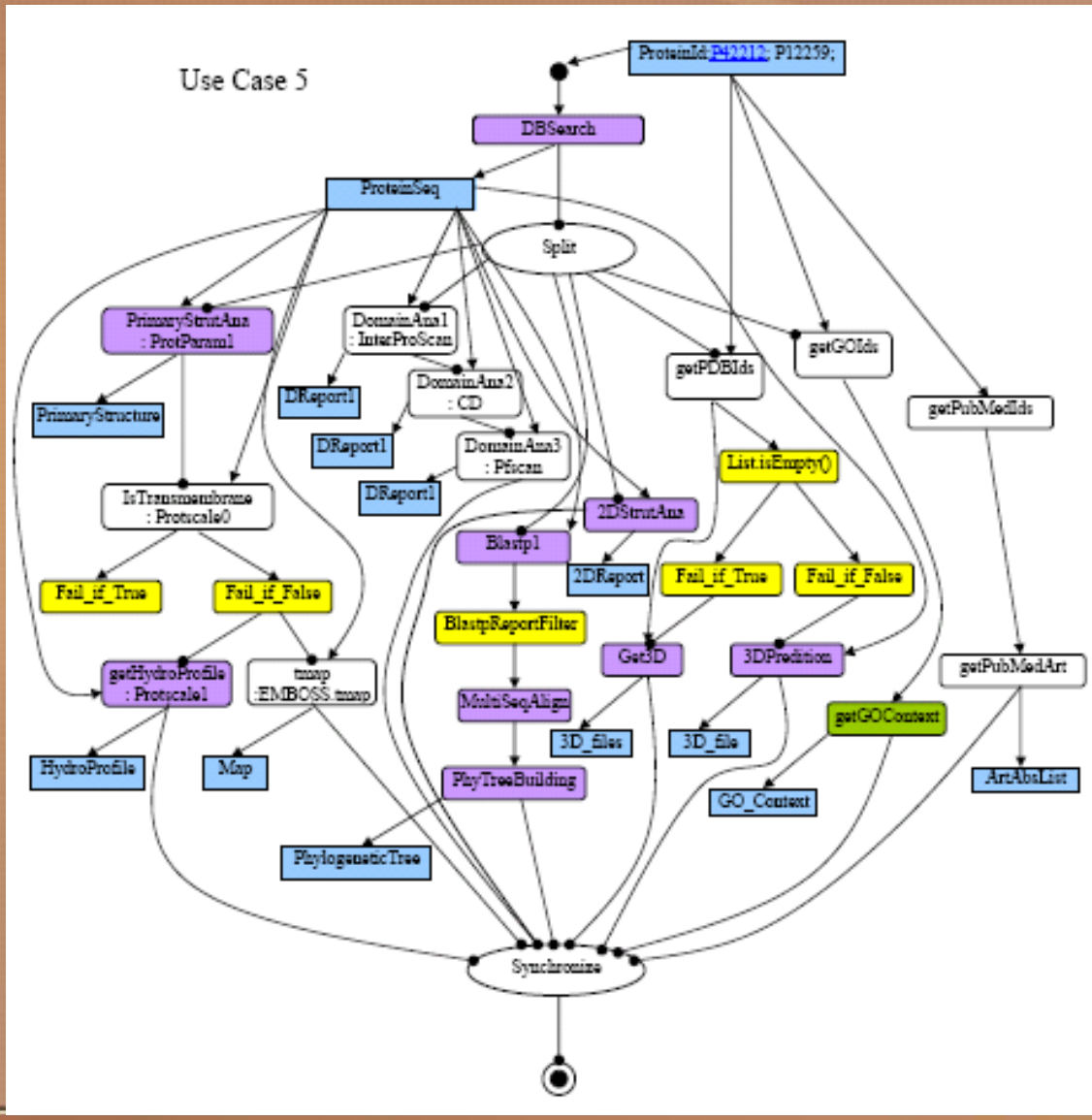
PhyLOG

- Allows design of workflows using a flow-chart like structure
  - nodes refers to data, concrete services, abstract service, or operators
- Automated configuration
  - mapping of abstract services to concrete services (discovery)
  - data binding
  - insertion of sequences of services
    - predefined templates (e.g., filtering, format transformation)
    - limited length automatically generated sequences of services
- Demonstrations
  - gene expression analysis for EST sentences
  - context analysis of a term in GO (myGrid workflow)
  - simple workflows for generation of phylogenetic trees

NIST UTD NM STATE UNIVERSITY

PhyLOG


Sequences
SequenceSize
SizeLimit
GreaterThan
FailIfTrue
FailIfFalse
Align
AlignedSequences


Use Case 5
ProteinId:P42212; P12259;
DBSearch
ProteinSeq
Split
PrimaryStrutAna : ProtParam1
DomainAna1 : InterProScan
DomainAna2 : CD
DomainAna3 : Pfscan
DReport1
DReport1
DReport1
PrimaryStructure
getPDBIds
getGOIds
getPubMedIds
List.isEmpty()
IsTransmembrane : Protscale0
2DStrutAna
Blastp1
2DReport
Fail_if_True
Fail_if_False
Fail_if_True
Fail_if_False
BlastpReportFilter
GetSD
3DPrediction
getPubMedArt
getHydroProfile : Protscale1
tmap :EMBOSS tmap
MultiSeqAlign
getGOContext
3D_files
3D_file
HydroProfile
Map
PhyTreeBuilding
GO_Content
ArtAbsList
PhylogeneticTree
Synchronize

Demonstration Projects

- Demonstration Projects
  - Projects to drive design, bootstrap development, and challenge technology
  - Sample selected projects
    - CDAO-mediated format transformations for high-value data sources (e.g., Pandit, KOGs)
      - In progress
    - Validation & Correction service for TreeBase
      - In progress
    - EGFams pipeline reimplementation
    - Workflow for evaluating multiple sequence alignment methods

NIST UTD NM STATE UNIVERSITY

PART II:

Problem Solving in Evolutionary Analysis with Web Services and DSLs

Workflows development

- Re-iterate the main ideas of workflow development environment
- Stress that here we will concentrate on the two key technologies
  - DSLs
  - composition/configuration

DSLs

- Design principles
- Development technology
- (perhaps stress link to CDAO)

Composition

- Web services (some intro)
- Semantic-based description (link again to CDAO?)
- composition problem
  - techniques
  - can mention also planning-based solutions