Processing Genome Data using Scalable Database Technology

Johann Christoph Freytag, Ph.D.
freytag@dbis.informatik.hu-berlin.de
http://www.dbis.informatik.hu-berlin.de

Stanford University, February 2004

My Background

PhD @ Harvard Univ.
Visiting Scientist, Microsoft Res. (2002)

- Starburst project, IBM Almaden Research Center (85-87)
- Visiting Scientist, Almaden Research Center (97/98)
- Visiting Scientist, IBM SVL (2001)

ERCR (European Computer Industry Research Centre), München (87-89)
DEC’s Database Technology Center, München (90-93)
“What’s the Meaning of Life”

DNA → RNA → Protein

- Transcription
- Translation
- Genomic „Transmitter“
- Messenger
- Gene product

Overview

- (Biological) Motivation/Problems
- Using Database Technology
  - Gene-EYe Integration-Platform
  - Data Cleansing
  - BLAST-Integration into GDB
  - In-and-Out-the-Database: Using Workflow for “dry” Experiments
- Summary
Processing Genome Data using Scalable Database Technology

View of Biological Areas

- Environment
- Diseases
- Experiments
- Pathways
- Life
- Evolution
- DNA Genome
- RNA Transcriptome
- Amino Acids Proteome

View of Data Data Source

- Environment
- Diseases
- Experiments
- OMIM
- Gene Ontology
- Evolution
- Taxonomy
- Brenda
- KEGG
- Life
- SWISS-PROT
- Interpro
- EMBL
- RefSeq
- LocusLink
- RNA
- EMBL (EST)
- SWISS-PROT
- Interpro
Complex Relationships

A graph depicting the relationships between 400+ biological data sources served by the EBI via SRS

More than 400 Data Sources on the WEB

Database Growth of EMBL (# of records)

Source: http://www3.ebi.ac.uk/Services/DBStats/

DBIS (our) Approach

Database Model of the Biological World

SwissProt

ESEMBLE

EMBL

KABAT
• (Biological) Motivation/Problems
• Using Scalable Database Technology
  – Gene-EYe Integration-Platform
  – Data Cleansing
  – BLAST-Integration into GDB
  – In-and-Out-the-Database: Using Workflow Concepts for “dry” Experiments
• Summary

Gene-EYe Integration-Platform

Vision

• Provide mechanisms for
  – unified handling of different data sources
  – data source integration
  – change management
  – user defined data preparation
• Provide
  – relevant tools for sequence manipulation and retrieval
  – work flow support for operation and administration
Gene-EYe Integration-Platform

The Big Picture

Genome Data Warehouse Layer (GDW Schema)

Genome Database Layer (GDB Schema)

Genome Data Store Layer (GDS Schema)

DATA

KNOCKLEDGE

CONTENT

Flat File Data -> Relational Entities (e.g. EMBL)

Relational Entities -> Biological Entities (e.g. Gene)

Biological Entities -> Biological Concepts (e.g. Life Cycle)

GDS: From Flat File to Database

Genome Data Store Layer (GDS Schema)

GDS Load Tools

GDS Admin Tools

Data Storage

Data Cleansing

Update/Admin

© Prof. J.C. Freytag, Verwendung
nur zum nicht-kommerziellen Gebrauch

6
The Data Import Pipeline - Revisited

Phase 1: Property Files
- Perl scripts
- Hand crafted

Phase 2: GEM Repository
- GEM1 Repository de.hui.dbis.geneeye.* (Java)
- Autogenerate from Metadata

1: CWM compliant GeneEYe Metadata Repository

Modeling the Maintenance Process

© Prof. J.C. Freytag, Verwendung
nur zum nicht-kommerziellen Gebrauch
GDB-Layer: From Data to Biology

Data Integration
Data Cleansing (Sem.)
Queries

Data Storage
Data Cleansing (Syn.)
Update/Admin

GDB Builder (IBM Clio?)

EMBL
SWALL
TAXO
InterPro
ENSEMBL

Gene
Protein
Transcript
Tissue
Variant

GDB Mapper (IBM Clio)

[Definition]

Defined by and in cooperation w/ domain experts

Genome Data Store Layer (GDS Schema)

Schema Mapping with Clio

Source Schema
User mapping
Target Schema

Clio

DB
SQL or XQuery

DB

with permission of Dr. Felix Naumann
IBM Almaden Research Center

© 2004 DBIS, Humboldt-Universität zu Berlin
Processing Genome Data
using Scalable Database Technology
Clio Features

- Schema Viewer
  - Visual mapping between schema elements
- Attribute Matcher
  - Intelligent suggestions of likely mappings
- Data Viewer
  - Data examples for mapping queries
- Queries
  - SQL, XSLT, Xquery
  - Use and adhere to source and target schema constraints

GDW: Providing Facts for Research

Genome Data Warehouse Layer (GDW Schema)

- GDW Miner
- GDB Explorer

Genome Database Layer (GDB Schema)

- Data Mining
- Ontology Mapping
- Process Simulation
- Data Integration
- Data Cleansing (Sem.)
- Queries
• (Biological) Motivation/Problems
• Using Scalable Database Technology
  – Gene-EYe Integration-Platform
  – Data Cleansing
  – BLAST-Integration into GDB
  – In-and-Out-the-Database: Using Workflow Concepts for “dry” Experiments

• Summary
Reliability-based Merging (cont.)

- Domain expert identifies reliable parts for merging
- Definition of a set of views for integration

Current work:
- Which are the relevant mismatch patterns?
- How to assess their relevance & importance?

\[ \text{e.g. MIN(} \]
BLAST: General Introduction

- Algorithm/Package: Similarity Search
- Developed by Altschul et al. (1990)
- Three Steps:
  1. Search for Word Pairs (Iseq, DSeq) of Length L on the Data Collection of Sequences above Threshold T
  2. Expansion of each Word Pair until the Value V of their Alignment is \( \Delta \) away from the local maximum
  3. Output of complete alignment (High-scoring Segment Pair, HSP), if Value(Alignment) > S

Output: Powerset of Alignments

BLAST UDF Implementation

- **Goal:** Using BLAST in SQL-statements
- **How?**
  - BLAST-UDF implemented as Table Function
  - Use in SQL Query
    ```sql
    SELECT *
    FROM TABLE( BLAST(<Parameter>, <Query Sequence>, <Comparison Sequence>) )
    ```
  - Each call returns a set of alignments over Sequences in the Database
Structure of UDB Table Function

- Implementation:
  - Mapping of program into calling structure for table functions
  - Communication between the different calls via scratchpad
  - scratchpad: Storage area which remains intact and unchanged between UDF calls
    - Storage of data structures for different steps
    - especially for output from postprocessing: SeqAlign

UDF BLAST

<table>
<thead>
<tr>
<th>FIRST</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initialize</td>
</tr>
<tr>
<td>OPEN</td>
</tr>
<tr>
<td>For all SEQUENCES</td>
</tr>
<tr>
<td>Alignment without gaps</td>
</tr>
<tr>
<td>Postprocessing</td>
</tr>
<tr>
<td>FETCH</td>
</tr>
<tr>
<td>For all ALIGNMENTS</td>
</tr>
<tr>
<td>Output of the results</td>
</tr>
<tr>
<td>CLOSE</td>
</tr>
<tr>
<td>Release data structures related with sequences</td>
</tr>
<tr>
<td>FINAL</td>
</tr>
<tr>
<td>Release global data structures</td>
</tr>
</tbody>
</table>

- (Biological) Motivation/Problems
- Using Scalable Database Technology
  - Gene-EYe Integration-Platform
  - BLAST-Integration into GDB
  - In-and-Out-the-Database: Using Workflow Concepts for "dry" Experiments
- Summary
The Challenge: Exon Skipping

One Gen with 100 Exons ⇒ $2^{100} \sim 10^{30}$ Variations

Exons within one Gene linearly combined (splicing) Used as Pattern for Protein Generation

Challenge: Exon Skipping

Do alternative fusion points new functional (i.e. biological meaningful) "patterns"?
Functional Genomics: Gain of New Insight

First Horizon: Simple Exon Skipping

Flow of Processing Steps

- Generate Exon Sequence
- Local Database (automatic)
- Remote Tool (Web Based)
- Find Similarity Search
- Check for Biological Validity

Supported by local DB
Implementation

Some facts...
- 60 days 100% load
- One splice form per minute
- So far: ca. 90,000 splice forms
- First biolog. meaningful results

Cooperations
- Cooperation with
  - Univ. of Jena (Rolf Backofen)
  - Berlin Center of Bioinformatics (BCB)
    - Charite, FU, Max-Planck-Institut (M. Vingron)
  - Industry: IBM, small companies, …

© Prof. Johann-Christoph Freytag, Verwendung nur zum nicht-kommerziellen Gebrauch
Summary

• Lesson learnt
  – Highly Dynamic Environment
    • Data: changes frequently
    • User: changes frequently
  – Provide a framework for
    • Date integration
    • Data processing
    • Data changes
    • Data dependencies…..
    • Meta data management

• Future Work
  – Query processing
    • Include domain knowledge
  – Data cleansing
  – Set of UDFs for biological data processing
  – Visualization of Data