GeWare 2.0 - On the way to a virtual repository

Toralf Kirsten

01.07.2008

What's GeWare?

- GeWare - Genetic Warehouse
- Platform for managing and analyzing large sets of experimental data generated by
  - high-thoughput expression experiments (Affymetrix)
  - custom-made CGH arrays (platform extension)

What's GeWare?

- GeWare - Genetic Warehouse
- Platform for managing and analyzing large sets of experimental data generated by
  - high-thoughput expression experiments (Affymetrix)
  - custom-made CGH arrays (platform extension)
- Generic approach to manage experiment annotations and import clinical data using annotation templates and controlled vocabularies

What's GeWare?

- GeWare - Genetic Warehouse
- Platform for managing and analyzing large sets of experimental data generated by
  - high-thoughput expression experiments (Affymetrix)
  - custom-made CGH arrays (platform extension)
- Generic approach to manage experiment annotations and import clinical data using annotation templates and controlled vocabularies
- Hybrid integration of web data using SRS

What's GeWare?

- GeWare - Genetic Warehouse
- Platform for managing and analyzing large sets of experimental data generated by
  - high-thoughput expression experiments (Affymetrix)
  - custom-made CGH arrays (platform extension)
- Generic approach to manage experiment annotations and import clinical data using annotation templates and controlled vocabularies
- Hybrid integration of web data using SRS
- Closed loop analysis workflow: Use of chip and gene/clone groups for easy and iterative analysis execution
- Data analysis by R BioCondutor (reuse of existing analysis software)

Content

What's GeWare
Requirements
GeWare 2.0 Approach
System Architecture
Kernel and Module Concepts
Conclusion & Future Work
Thesis Topics

New Requirements

- Requirements born by upcoming very (!) large scale projects and clinical trials
- **New** experimental data types generated by new high-throughput array techniques, e.g., tiling ~ exon ~ snp arrays
- **New** chip producers, e.g., Illumina
- Integration of **third party** data management solutions, e.g. BC|SNPMax
- Generic exchange of clinical data, e.g., by using XML
- Need for a **generic** approach to store all types of object groups
- User-right-management: rights per user on data and function

New Requirements cont.

- Flexible analysis integration (upload and execution of R analysis scripts)
- Move of (long running) analysis jobs to dedicated analysis servers
- Storage and return (on demand) of RData objects for every analysis
- Flexible analysis annotation (use of ontologies?)
- **Versioned integration** of web-data
- Web-GUI using Web 2.0 techniques
- Fat client on desktop (e.g., Rich Client)

GeWare 2.0

- Need for a redesign of GeWare ($\leq$1.0) $\rightarrow$ GeWare 2.0
- **GeWare 2.0**: Mediator on warehouses
  - own warehouse (ext. DB schema of GeWare 1.0)
  - other sources of high-throughput data

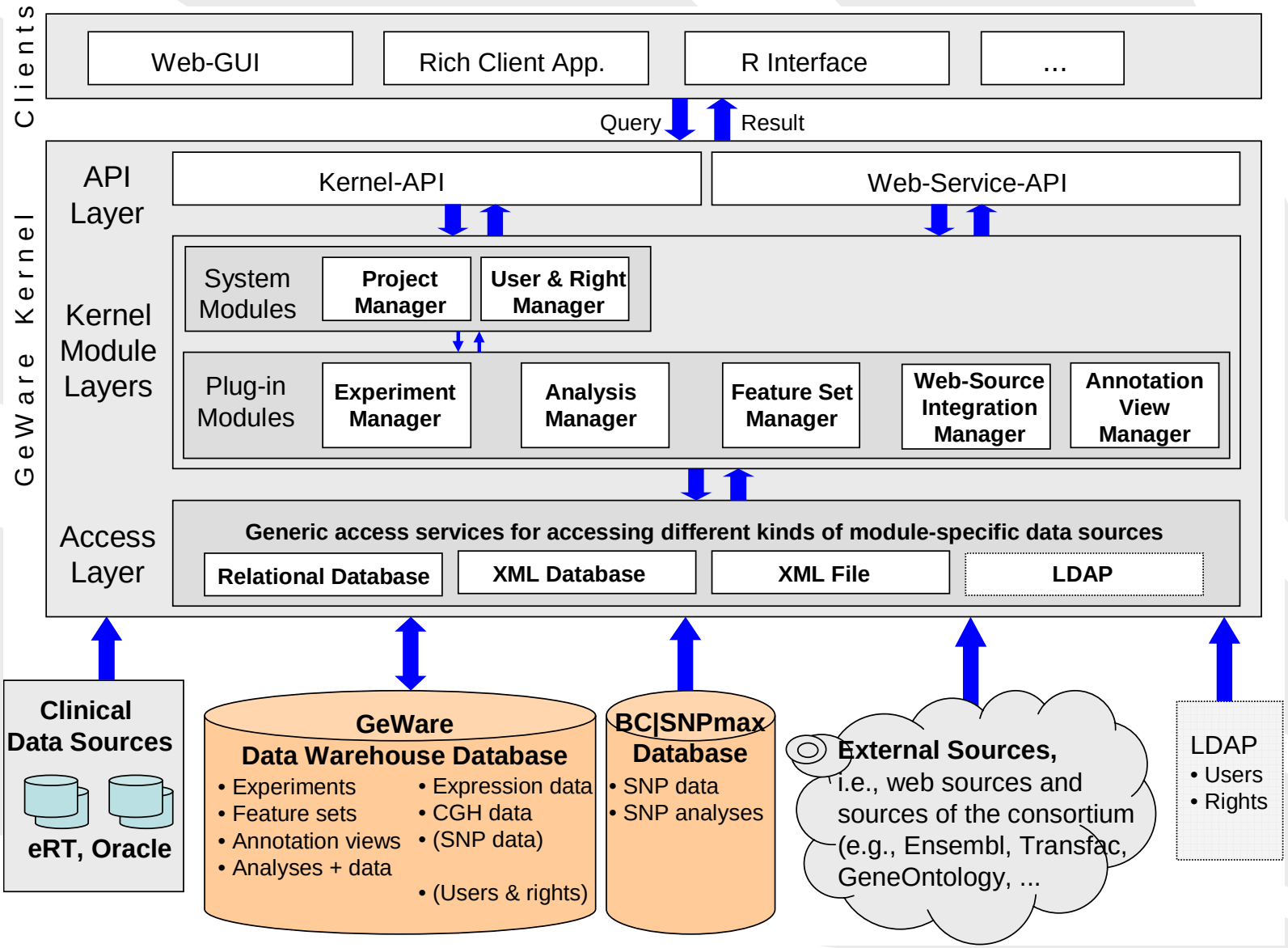
GeWare 2.0

- Need for a redesign of GeWare ($\leq$1.0) $\rightarrow$ GeWare 2.0
- **GeWare 2.0**: Mediator on warehouses
  - own warehouse (ext. DB schema of GeWare 1.0)
  - other sources of high-throughput data
- **Generic** source access layer
  - independence of source-specific query syntax
  - portability

GeWare 2.0

- Need for a redesign of GeWare ($\leq$1.0) $\rightarrow$ GeWare 2.0
- **GeWare 2.0**: Mediator on warehouses
  - own warehouse (ext. DB schema of GeWare 1.0)
  - other sources of high-throughput data
- **Generic** source access layer
  - independence of source-specific query syntax
  - portability
- **Modularization**: Kernel and task-specific modules
  - system vs. plug-in modules
  - flexible module development (integrate a new or replace an existing module) by using a defined abstract module interface

System Architecture



Generic Source Access Layer

- Initialization by a set of data source connections
- Transparent data source access by kernel and modules
- Connection pooling for fast and parallel access
  - Problem: Which modules can store / retrieve data from which data source?
  - Solution: source-specific metadata

source-metadata(source id, module name, schema version)

- Mount (and unmount) data sources on system's runtime
- Further problems:
  - Uniqueness of object identifiers
  - Selection of insert source

Uniqueness of Object Identifiers

- Problem: No unique object identifiers when multiple sources are used for same type of data
- Solution: Encoding and decoding of source-specific object identifiers

$$f : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$$

$$f(id_{Src\,k-Obj}, k) = id_{Sys-Obj} = id_{Src\,k-Obj} * c + k$$

$k \in \mathbb{N} :=$ unique source identifier

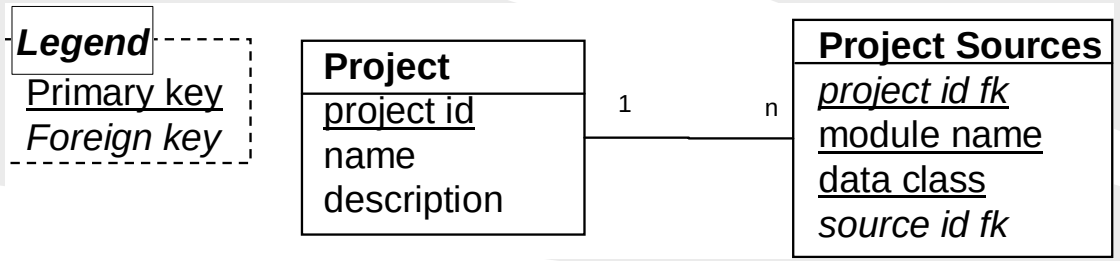
$c = 10^e, e \in \mathbb{N}, e > 0 (e.g., c = 1000)$

$$f^{-1}_{Src}(id_{Sys-Obj}) = k = id_{Sys-Obj} \mod c$$

$$f^{-1}_{Obj}(id_{Sys-Obj}, k) = id_{Src\,k-Obj} = (id_{Sys-Obj} - k)/c$$

Project Manager

- Project
  - typically, singular (planned) process for a specified time
  - in GeWare: Container for storing every type of data; project semantics keeps in user's responsibility
- Problem: Selection of an insert source
- Solution:
  - Association of each project and module with a singluar data source as selection for insert operation
  - read, update and delete for all other data sources



Experiment Manager

- **Experiment** = Collection of
  - metadata (name, type, etc.) about high-throughput arrays, e.g., expression, exon, ... arrays
  - array-based experiment annotation

Experiment Manager

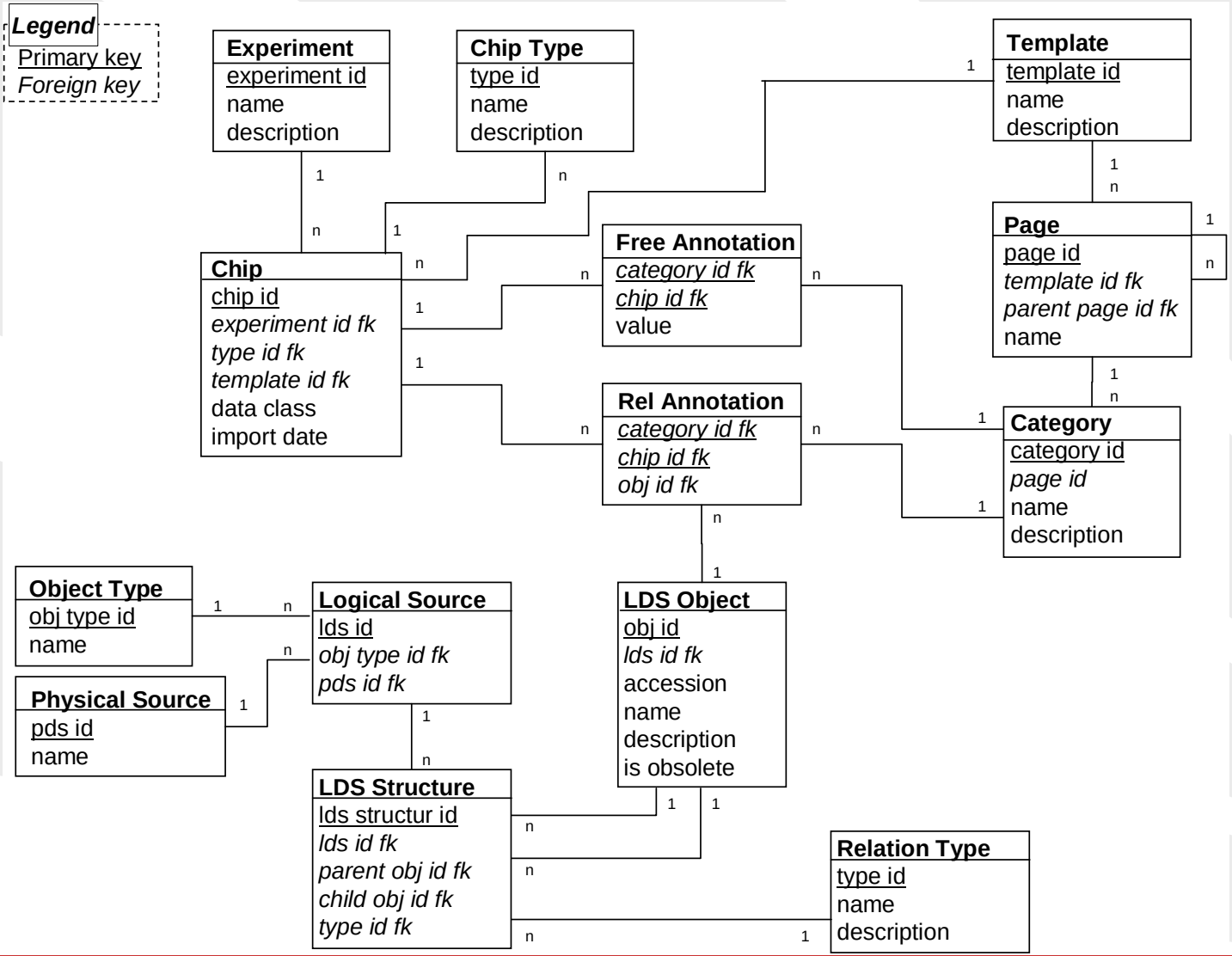
- **Experiment** = Collection of
  - metadata (name, type, etc.) about high-throughput arrays, e.g., expression, exon, ... arrays
  - array-based experiment annotation
- Experiment annotation on basis of **annotation templates**
  - set of hierarchically organized categories for which values can be captured
  - can be organized on pages

Experiment Manager

- **Experiment** = Collection of
  - metadata (name, type, etc.) about high-throughput arrays, e.g., expression, exon, ... arrays
  - array-based experiment annotation
- Experiment annotation on basis of **annotation templates**
  - set of hierarchically organized categories for which values can be captured
  - can be organized on pages
- Input values
  - manual input
  - selected **ontology** concepts (no flat vocabularies)

Experiment Manager cont.

Module-specific Database Schema Portion



Feature Set Manager

- Feature = **generic object** representing an object of interest, such as genes, proteins but also chips
  - analysis result
  - user-specified

Feature Set Manager

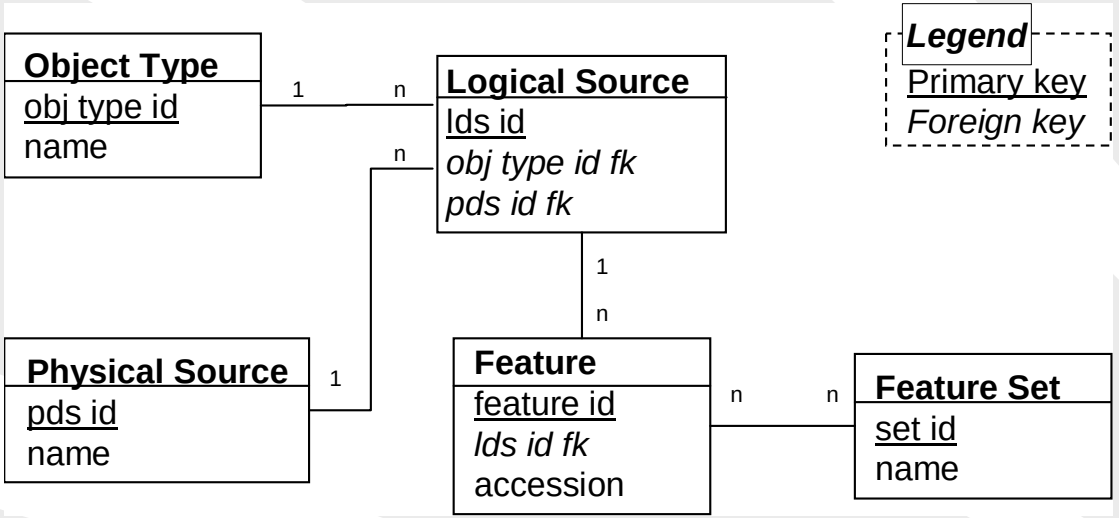
- Feature = **generic object** representing an object of interest, such as genes, proteins but also chips
  - analysis result
  - user-specified
- **Feature Set** = user-specified / pre-specified set of features
- Capture
  - semantics by using a meaningful object type and
  - data lineage given by a physical source

Feature Set Manager

- Feature = **generic object** representing an object of interest, such as genes, proteins but also chips
  - analysis result
  - user-specified
- **Feature Set** = user-specified / pre-specified set of features
- Capture
  - semantics by using a meaningful object type and
  - data lineage given by a physical source
- Methods for
  - retrieval and im-/export
  - set manipulation, e.g., union, intersect, diff, majority

Feature Set Manager cont.

Module-specific Database Schema Portion



Annotation View Manager

- Annotation = Description, explaination of specific object by using an attribute set
- Annotation view = defined set of attributes and corresponding values describing a primary (biological) object

Annotation View Manager

- Annotation = Description, explaination of specific object by using an attribute set
- Annotation view = defined set of attributes and corresponding values describing a primary (biological) object

Primary object

Annotation (set of describing attributes)

Probe Set ID	Gene Title	Gene Symbol	go biological process term	go molecular function term	go cellular component term	Pathway
1000_at	mitogen-activated protein kinase 3	MAPK3	protein amino acid phosphorylation protein amino acid phosphorylation cell cycle	nucleotide binding protein kinase activity protein serine/threonine kinase activity MAP kinase activity MAP kinase activity protein binding ATP binding		MAPK_Cascade S1P_Signaling TGF_Beta_Signaling_Pathway

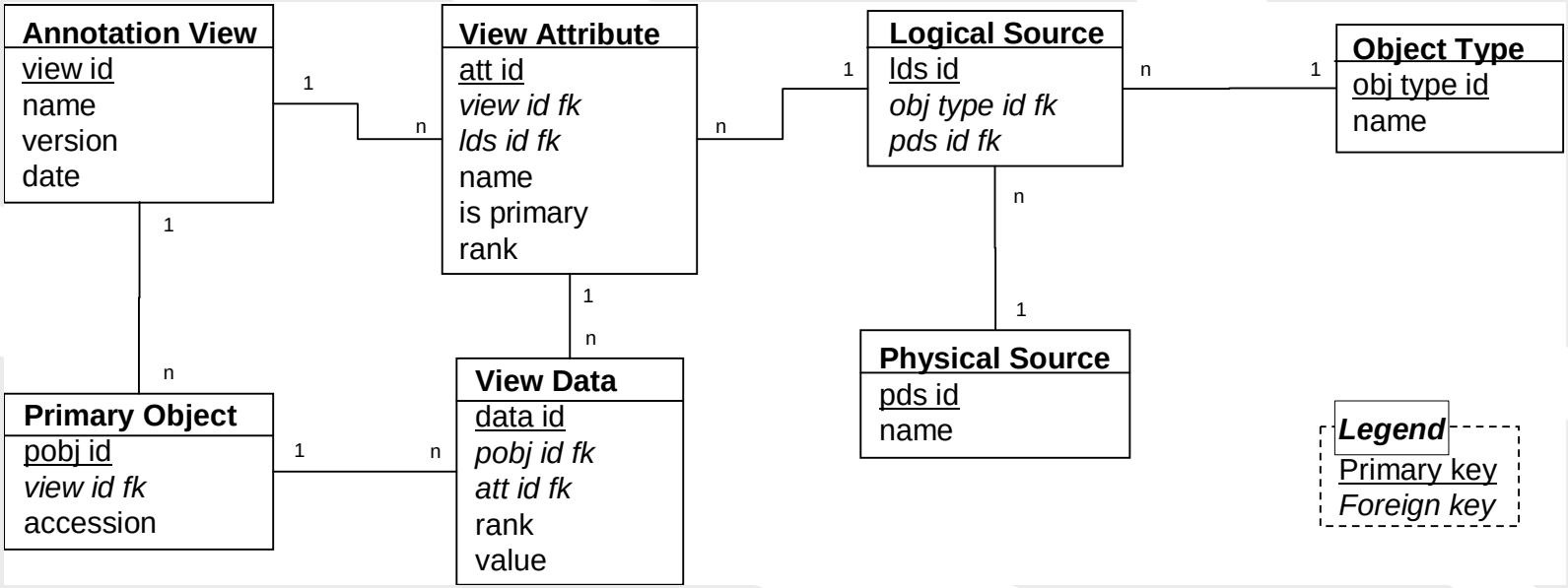
Annotation View Manager

- Annotation = Description, explaination of specific object by using an attribute set
- Annotation view = defined set of attributes and corresponding values describing a primary (biological) object

- Capture
  - semantics by using a meaningful object type and
  - data lineage given by a physical source

Annotation View Manager cont.

Module-specific Database Schema Portion



Web-Source Integration Manager

- Current idea: Use of BioFuice to integrate data from different source
- Implementation of integration workflows (iFuice scripts)

Web-Source Integration Manager

- Current idea: Use of BioFuice to integrate data from different source
- Implementation of integration workflows (iFuice scripts)
- Problems:
  - versioning of physical and logical sources
  - generation of annotation views

Web-Source Integration Manager

- Current idea: Use of BioFuice to integrate data from different source
- Implementation of integration workflows (iFuice scripts)
- Problems:
  - versioning of physical and logical sources
  - generation of annotation views
- Solution (First Approach)
  - versioning: physical source name = source name + version
  - view generation: new method genView() in BioFuice

$$view = genView(M_1, \ldots, M_n, A_{s_i})$$

where it exists a compositional path from mapping M_1 to M_n and set of attributes A_{s_i} from sources $s_i \in (domain(M_j) \in \{M_1, \ldots, M_n\} | range(M_j) \in \{M_1, \ldots, M_n\})$

Conclusion & Future Work

- Need for a redesign of GeWare based on new requirements
- GeWare 2.0: Mediator of warehouses
- Modularization: Kernel and task-specific modules
- Current state: Concept and implementation of system architecture, kernel and most modules

Conclusion & Future Work

- Need for a redesign of GeWare based on new requirements
- GeWare 2.0: Mediator of warehouses
- Modularization: Kernel and task-specific modules
- Current state: Concept and implementation of system architecture, kernel and most modules
- Future work
  - Need to implement: Experiment & Web-Source Integration Manager
  - Need a detailed concept for analysis handling, monitoring, distribution (e.g., by using available Grid infrastructure), and annotation

Topics for Diploma/Master/Bachelor Thesis

- Web-GUI (re-)design
- Rich client application
- Application allowing to configure the clinical data exchange
- Analysis management and distribution

GeWare-Coders-Club

- What's GeWare
- Requirements
- GeWare 2.0 Approach
- System Architecture
- Kernel and Module Concepts
- Conclusion & Future Work
- Thesis Topics

”Gehirnsturm am Freitagabend”

- What’s GeWare
- Requirements
- GeWare 2.0 Approach
- System Architecture
- Kernel and Module Concepts
- Conclusion & Future Work
- Thesis Topics