EuPathDB Infrastructure Document

Overview

This document provides a high-level overview of the software infrastructure utilized by the EuPathDB BRC to load, integrate and provide data to users. The links provide details for each of the components summarized here. If you would like a PDF of this information, please download from the following link.

- EuPathDB_Infrastructure.pdf

Data Loading and Database Schema

We use the **G**enomics **U**nified **S**chema (GUS) database schema and data loading infrastructure available at GusDB.org. This includes not only a comprehensive database schema for integrating and representing genomic and functional (or post) genomic data but also tools for loading said data into that system. We have made some extenstions to the schema and tools for EuPathDB specific purposes primarily to generate de-normalized views of the data for query optimization purposes.

Our data are all stored in Oracle11g databases. Our software infrastructure also supports PostgreSQL but we have some Oracle specific SQL constructs in our model that would need to be changed in order to run successfully in PostgreSQL.

We load all data using an in house engineered workflow sytem called ReFlow. Documentation and code for ReFlow can be found on the ReFlow google code repository. Briefly, ReFlow is engineered to be an efficient graph based workflow system any step of which has the ability to be undone and subsequently rerun with updated data. This was a significant requirement as it enables us to undo entire genomes when the annotation or underlying sequence changes. This results in automated removal of all data dependent on that genome and when it is re-run with the new annotation, all dependent data are recomputed and reloaded automatically, thus greatly improving our ability to keep these complex databases up-to-date. A significant component of our dataloading system is our ability to automate the running of large compute tasks on compute clusters located at the University of Pennsylvania and the University of Georgia. The ReFlow workflow sytem utilizes another piece of software developed at the University of Pennsylvania to schedule, manage and monitor running tasks called DistribJob. DistribJob distributes tasks generated from a large input dataset such as a set of sequences to compute nodes in a cluster for analysis and retrieves and collates the results in an efficient manner.

- GUS Schema and Tools
  - GUS Schema
  - GUS developers guide
- ReFlow Workflow System
- DistribJob Cluster Task Management

Web Presentation System and User Interfaces

Our websites are based on code that we developed and have released to the community called the Strategies-WDK (Strategies Web Development Kit) which enables the graphical strategies search system. You can download the software and see documentation for this toolkit at Strategies-WDK google code website. This toolkit enables us to represent our data as an XML model which is then turned into the web interfaces that are presented to users using these tools.

- [Strategies-WDK Website](#)
- [Strategies-WDK Documentation](#)

System Hardware and Third Party Software

EuPathDB maintains redundant database and content web servers at the University of Pennsylvania and the University of Georgia to minimize interruptions for our users during maintenance periods. Additionally EuPathDB compute and data loading servers are located at the University of Pennsylvania.

Server configurations are coordinated and deployed through Puppet automation software (http://puppetlabs.com/). Custom infrastructure software is versioned and deployed through standard RPM/YUM mechanisms. When appropriate, software builds are automated using Jenkins Continuous Integration Server (http://jenkins-ci.org/) and deloyed into an internal Sonatype Nexus (http://www.sonatype.org/nexus/) repository.

System infrastructure statistics (CPU load, I/O, etc) are gathered with colletd (http://collectd.org/) and in-house applications and feed to Graphite (http://graphite.wikidot.com/) for human review. Nagios (http://www.nagios.org/) provides notifications of system degradations.

Both Universities also maintain large compute clusters that are heavily utilized by EuPathDB in order to analyze and load incoming data in a timely fashion. The linked document below describes our actual hardware and includes a list of third-party software required in order to analyze, load and present data via our websites.

- [Hardware and Third Party Software](#)

Community Interactions and Data Submission Policies

EuPathDB serves a global scientific community that demands direct active support and community involvement. EuPathDB outreach activities include:

- Organizing and running hands on training workshops and webinars ([Google Map](#)).
- Developing educational material in the form of [exercises](#) and [online tutorials](#).
- Responding to support emails for users who contact us directly by clicking the "Contact Us" links in the header or footer of any EuPathDB webpage (average response time is 48 hours).
- Holding semi-annual meetings with our [Scientific Working Group](#) to discuss high-level goals and deliverables for the project.
- Holding bimonthly conference calls with our [Scientific Advisory Groups](#) (these are organism specific, one per component site with TriTrypDB separated into three) to learn about new datasets, set priorities and share with the community current activities.
- Attending national and international meetings with active participation in the form of posters, presentations or help desks.
- Authoring [peer reviewed manuscripts](#).
- Maintaining active social media presence in the form of a [FaceBook page](#) and [Twitter feed](#).
- Providing a clear [data handling and release policy](#) to investigators to encourage submission of data prepublication.

GUS Schema Browser

Table-Centric View | Category-Centric View

Table	Superclass	
Sequence and Features		
NA Sequence		Return to Super Category
DoTS::Assembly	DoTS::NASequence	View Table | View Documentation | Edit
DoTS::DbRefNASequence		View Table | View Documentation | Edit
DoTS::ExternalNASequence	DoTS::NASequence	View Table | View Documentation | Edit
DoTS::NASequence		View Table | View Documentation | Edit
DoTS::SequencePiece		View Table | View Documentation | Edit
DoTS::SplicedNASequence	DoTS::NASequence	View Table | View Documentation | Edit
DoTS::VirtualSequence	DoTS::NASequence	View Table | View Documentation | Edit
AA Sequence		Return to Super Category
DoTS::AASequence		View Table | View Documentation | Edit
DoTS::DbRefPfamEntry		View Table | View Documentation | Edit
DoTS::ExternalAASequence	DoTS::AASequence	View Table | View Documentation | Edit
DoTS::MotifAASequence	DoTS::AASequence	View Table | View Documentation | Edit
DoTS::NRDBEntry		View Table | View Documentation | Edit
DoTS::PfamEntry		View Table | View Documentation | Edit
DoTS::TranslatedAASequence	DoTS::AASequence	View Table | View Documentation | Edit
DoTS::TrivialTranslation	DoTS::AASequence	View Table | View Documentation | Edit
NA Sequence Features		Return to Super Category
DoTS::AlleleFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::BindingSiteFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::ChromosomeElementFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::CollectionNAFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::DNARegulatory	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::DNAStructure	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::ExonFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::GeneFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::HexamerFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::Immunoglobulin	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::InflectionPointFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::IntronFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::LowComplexityNAFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::Miscellaneous	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::NAFeature		View Table | View Documentation | Edit
DoTS::PolyAFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::PromoterFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::ProteinFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::Repeats	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::ReplicationOriginFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::RestrictionFragmentFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::RNAFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::RNARegulatoryFeature	DoTS::NAFeature	View Table | View Documentation | Edit

DoTS::RNAStructure	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::RNAType	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::SAGETagFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::ScaffoldGapFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::SeqVariation	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::Source	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::SpliceSiteFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::STS	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::TandemRepeatFeature	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::Transcript	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::TransposableElement	DoTS::NAFeature	View Table | View Documentation | Edit
DoTS::UTRFeature	DoTS::NAFeature	View Table | View Documentation | Edit
AA Sequence Features		Return to Super Category
DoTS::AAFeature		View Table | View Documentation | Edit
DoTS::AASeqVariation	DoTS::AAFeature	View Table | View Documentation | Edit
DoTS::DomainFeature	DoTS::AAFeature	View Table | View Documentation | Edit
DoTS::EpitopeFeature	DoTS::AAFeature	View Table | View Documentation | Edit
DoTS::LowComplexityAAFeature	DoTS::AAFeature	View Table | View Documentation | Edit
DoTS::MassSpecFeature	DoTS::AAFeature	View Table | View Documentation | Edit
DoTS::PlasmoAPFeature	DoTS::AAFeature	View Table | View Documentation | Edit
DoTS::PostTranslationalModFeature	DoTS::AAFeature	View Table | View Documentation | Edit
DoTS::PredictedAAFeature	DoTS::AAFeature	View Table | View Documentation | Edit
DoTS::RepeatRegionAAFeature	DoTS::AAFeature	View Table | View Documentation | Edit
DoTS::SecondaryStructureAAFeature	DoTS::AAFeature	View Table | View Documentation | Edit
DoTS::SignalPeptideFeature	DoTS::AAFeature	View Table | View Documentation | Edit
DoTS::TertiaryStructureAAFeature	DoTS::AAFeature	View Table | View Documentation | Edit
DoTS::TranslatedAAFeatSeg		View Table | View Documentation | Edit
DoTS::TranslatedAAFeature	DoTS::AAFeature	View Table | View Documentation | Edit
Feature Relations		Return to Super Category
DoTS::AALocation		View Table | View Documentation | Edit
DoTS::DbRefNAFeature		View Table | View Documentation | Edit
DoTS::ExonLocation		View Table | View Documentation | Edit
DoTS::FeatureName		View Table | View Documentation | Edit
DoTS::GeneFeatureSAGETagLink		View Table | View Documentation | Edit
DoTS::NAFeatRelationshipType		View Table | View Documentation | Edit
DoTS::NAFeatureComment		View Table | View Documentation | Edit
DoTS::NAFeatureRelationship		View Table | View Documentation | Edit
DoTS::NALocation		View Table | View Documentation | Edit
DoTS::RNAFeatureExon		View Table | View Documentation | Edit
GenBank Sequence Record		Return to Super Category
DoTS::AAComment		View Table | View Documentation | Edit
DoTS::AAEntry		View Table | View Documentation | Edit
DoTS::AAGene		View Table | View Documentation | Edit
DoTS::AAGeneSynonym		View Table | View Documentation | Edit
DoTS::AASequenceKeyword		View Table | View Documentation | Edit
DoTS::AASequenceOrganelle		View Table | View Documentation | Edit

DoTS::AASequenceRef		View Table | View Documentation | Edit
DoTS::AASequenceTaxon		View Table | View Documentation | Edit
DoTS::CommentName		View Table | View Documentation | Edit
DoTS::Keyword		View Table | View Documentation | Edit
DoTS::NAComment		View Table | View Documentation | Edit
DoTS::NAEntry		View Table | View Documentation | Edit
DoTS::NAFeatureNAGene		View Table | View Documentation | Edit
DoTS::NAFeatureNAProtein		View Table | View Documentation | Edit
DoTS::NAFeatureNAPT		View Table | View Documentation | Edit
DoTS::NAGene		View Table | View Documentation | Edit
DoTS::NAPrimaryTranscript		View Table | View Documentation | Edit
DoTS::NAProtein		View Table | View Documentation | Edit
DoTS::NASeqCytoLocation		View Table | View Documentation | Edit
DoTS::NASequenceKeyword		View Table | View Documentation | Edit
DoTS::NASequenceOrganelle		View Table | View Documentation | Edit
DoTS::NASequenceRef		View Table | View Documentation | Edit
DoTS::Organelle		View Table | View Documentation | Edit
DoTS::SecondaryAccs		View Table | View Documentation | Edit
Similarity		*Return to Super Category*
DoTS::BLATAlignment		View Table | View Documentation | Edit
DoTS::BLATAlignmentQuality		View Table | View Documentation | Edit
DoTS::ConsistentAlignment		View Table | View Documentation | Edit
DoTS::Similarity		View Table | View Documentation | Edit
DoTS::SimilaritySpan		View Table | View Documentation | Edit
Assembly		*Return to Super Category*
DoTS::AssemblyAnatomyPercent		View Table | View Documentation | Edit
DoTS::AssemblySequence		View Table | View Documentation | Edit
DoTS::AssemblySequenceSNP		View Table | View Documentation | Edit
DoTS::AssemblySNP		View Table | View Documentation | Edit
DoTS::GeneTrapAssembly		View Table | View Documentation | Edit
DoTS::MergeSplit		View Table | View Documentation | Edit
Motifs		*Return to Super Category*
DoTS::Motif		View Table | View Documentation | Edit
DoTS::MotifRejectionReason		View Table | View Documentation | Edit
DoTS::RejectedMotif		View Table | View Documentation | Edit

Function

Central Dogma		*Return to Super Category*
DoTS::Gene		View Table | View Documentation | Edit
DoTS::GeneCategory		View Table | View Documentation | Edit
DoTS::GeneChromosomalLocation		View Table | View Documentation | Edit
DoTS::GeneInstance		View Table | View Documentation | Edit
DoTS::GeneInstanceCategory		View Table | View Documentation | Edit
DoTS::GeneSynonym		View Table | View Documentation | Edit
DoTS::Protein		View Table | View Documentation | Edit
DoTS::ProteinCategory		View Table | View Documentation | Edit
DoTS::ProteinInstance		View Table | View Documentation | Edit

DoTS::ProteinInstanceCategory		View Table | View Documentation | Edit
DoTS::ProteinProperty		View Table | View Documentation | Edit
DoTS::ProteinPropertyType		View Table | View Documentation | Edit
DoTS::ProteinProteinCategory		View Table | View Documentation | Edit
DoTS::ProteinSynonym		View Table | View Documentation | Edit
DoTS::RNA		View Table | View Documentation | Edit
DoTS::RNAAnatomy		View Table | View Documentation | Edit
DoTS::RNAAnatomyLOE		View Table | View Documentation | Edit
DoTS::RNACategory		View Table | View Documentation | Edit
DoTS::RNAIConstruct		View Table | View Documentation | Edit
DoTS::RNAInstance		View Table | View Documentation | Edit
DoTS::RNAInstanceCategory		View Table | View Documentation | Edit
DoTS::RNAIPhenotype		View Table | View Documentation | Edit
DoTS::RNARNACategory		View Table | View Documentation | Edit
DoTS::RNASecondaryStructure	DoTS::NAFeature	View Table | View Documentation | Edit
Paralog and Family		*Return to Super Category*
DoTS::AAFamilyExperiment		View Table | View Documentation | Edit
DoTS::AAParalogExperiment	DoTS::AASeqGroupExperiment	View Table | View Documentation | Edit
DoTS::AAParalogGroup	DoTS::AASequenceGroup	View Table | View Documentation | Edit
DoTS::AASequenceFamily		View Table | View Documentation | Edit
DoTS::AASequenceGroupFamily		View Table | View Documentation | Edit
DoTS::InParalogExperiment	DoTS::SeqGroupExperiment	View Table | View Documentation | Edit
DoTS::InParalogGroup	DoTS::SequenceGroup	View Table | View Documentation | Edit
DoTS::OrthologExperiment	DoTS::SeqGroupExperiment	View Table | View Documentation | Edit
DoTS::OrthologGroup	DoTS::SequenceGroup	View Table | View Documentation | Edit
DoTS::OutParalogExperiment	DoTS::SeqGroupExperiment	View Table | View Documentation | Edit
DoTS::OutParalogGroup	DoTS::SequenceGroup	View Table | View Documentation | Edit
DoTS::SeqGroupExperiment		View Table | View Documentation | Edit
DoTS::SequenceFamily		View Table | View Documentation | Edit
DoTS::SequenceFamilyExperiment		View Table | View Documentation | Edit
DoTS::SequenceGroup		View Table | View Documentation | Edit
DoTS::SequenceGroupFamily		View Table | View Documentation | Edit
DoTS::SequenceSequenceGroup		View Table | View Documentation | Edit
Sequence Ortholog, Paralog, Family		*Return to Super Category*
AA Ortholog		*Return to Super Category*
DoTS::AAOrthologExperiment	DoTS::AASeqGroupExperiment	View Table | View Documentation | Edit
DoTS::AAOrthologGroup	DoTS::AASequenceGroup	View Table | View Documentation | Edit
DoTS::AASeqGroupExperiment		View Table | View Documentation | Edit
DoTS::AASequenceGroup		View Table | View Documentation | Edit
DoTS::AASequenceSequenceGroup		View Table | View Documentation | Edit
DoTS::BestSimilarityPair		View Table | View Documentation | Edit
Mapping		*Return to Super Category*
DoTS::EndSequencePairMap		View Table | View Documentation | Edit
DoTS::EPCR		View Table | View Documentation | Edit
DoTS::FingerprintClone		View Table | View Documentation | Edit
DoTS::FingerprintCloneContig		View Table | View Documentation | Edit

DoTS::FingerprintCloneMarker		View Table | View Documentation | Edit
DoTS::FingerprintContig		View Table | View Documentation | Edit
DoTS::FingerprintMap		View Table | View Documentation | Edit
DoTS::MicroSatelliteMap		View Table | View Documentation | Edit
DoTS::OpticalMapAlignment		View Table | View Documentation | Edit
DoTS::OpticalMapAlignmentSpan		View Table | View Documentation | Edit
DoTS::OpticalMapFragment		View Table | View Documentation | Edit
DoTS::RHMap		View Table | View Documentation | Edit
DoTS::RHMapMarker		View Table | View Documentation | Edit
DoTS::RHMarker		View Table | View Documentation | Edit
Clones		Return to Super Category
DoTS::AnatomyLibrary		View Table | View Documentation | Edit
DoTS::AnatomyLOE		View Table | View Documentation | Edit
DoTS::Clone		View Table | View Documentation | Edit
DoTS::CloneInSet		View Table | View Documentation | Edit
DoTS::CloneSet		View Table | View Documentation | Edit
DoTS::DevStageLibrary		View Table | View Documentation | Edit
DoTS::EST		View Table | View Documentation | Edit
DoTS::Library		View Table | View Documentation | Edit
Interaction		Return to Super Category
DoTS::Complex		View Table | View Documentation | Edit
DoTS::ComplexAnatomy		View Table | View Documentation | Edit
DoTS::ComplexComponent		View Table | View Documentation | Edit
DoTS::ComplexType		View Table | View Documentation | Edit
DoTS::EffectorActionType		View Table | View Documentation | Edit
DoTS::Interaction		View Table | View Documentation | Edit
DoTS::InteractionInteractionLOE		View Table | View Documentation | Edit
DoTS::InteractionLOE		View Table | View Documentation | Edit
DoTS::InteractionType		View Table | View Documentation | Edit
DoTS::Pathway		View Table | View Documentation | Edit
DoTS::PathwayInteraction		View Table | View Documentation | Edit
DoTS::RowSet		View Table | View Documentation | Edit
DoTS::RowSetMember		View Table | View Documentation | Edit
GO Association		Return to Super Category
DoTS::GOAssociation		View Table | View Documentation | Edit
DoTS::GOAssociationInstance		View Table | View Documentation | Edit
DoTS::GOAssociationInstanceLOE		View Table | View Documentation | Edit
DoTS::GOAssocInstEvidCode		View Table | View Documentation | Edit
Variation		Return to Super Category
DoTS::Allele		View Table | View Documentation | Edit
DoTS::AlleleComplementation		View Table | View Documentation | Edit
DoTS::AlleleInstance		View Table | View Documentation | Edit
DoTS::AllelePhenotype		View Table | View Documentation | Edit
DoTS::AllelePhenotypeClass		View Table | View Documentation | Edit
DoTS::Complementation		View Table | View Documentation | Edit
Raw Mass Spectrometry Results		Return to Super Category

Protein Identification (From Mass Spec Analysis)		Return to Super Category
Transcription Regulation		
Regulation Framework		Return to Super Category
Regulatory Moieties		Return to Super Category
TESS::Moiety		View Table | View Documentation | Edit
TESS::MoietyComplex	TESS::Moiety	View Table | View Documentation | Edit
TESS::MoietyHeteroDimer	TESS::Moiety	View Table | View Documentation | Edit
TESS::MoietyInstance		View Table | View Documentation | Edit
TESS::MoietyMember		View Table | View Documentation | Edit
TESS::MoietyMonomer	TESS::Moiety	View Table | View Documentation | Edit
TESS::MoietyMultimer	TESS::Moiety	View Table | View Documentation | Edit
TESS::MoietySynonym		View Table | View Documentation | Edit
Regulatory Activities		Return to Super Category
TESS::Activity		View Table | View Documentation | Edit
TESS::ActivityConditions		View Table | View Documentation | Edit
TESS::ActivityGeneRegulation	TESS::Activity	View Table | View Documentation | Edit
TESS::ActivityInferenceSource		View Table | View Documentation | Edit
TESS::ActivityModel		View Table | View Documentation | Edit
TESS::ActivityProteinDnaBinding	TESS::Activity	View Table | View Documentation | Edit
Comments		Return to Super Category
TESS::Note		View Table | View Documentation | Edit
Models		Return to Super Category
TESS::MarkovChainObs		View Table | View Documentation | Edit
TESS::Model		View Table | View Documentation | Edit
TESS::ModelMarkovChain	TESS::Model	View Table | View Documentation | Edit
TESS::ModelSbcgGrammar	TESS::Model	View Table | View Documentation | Edit
Multinomial Models		Return to Super Category
TESS::MultinomialLabel		View Table | View Documentation | Edit
TESS::MultinomialLabelSet		View Table | View Documentation | Edit
TESS::MultinomialObservation		View Table | View Documentation | Edit
TESS::MultinomialObservationSet		View Table | View Documentation | Edit
Families of Similar Models		Return to Super Category
Occam's Razor		Return to Super Category
High Volume Genome Annotation		Return to Super Category
TESS::ArchipelagoOfInterest		View Table | View Documentation | Edit
TESS::IslandOfInterest		View Table | View Documentation | Edit
TESS::PredictedBindingSite		View Table | View Documentation | Edit
TESS::RegionOfInterest		View Table | View Documentation | Edit
Training Sets		Return to Super Category
TESS::Footprint		View Table | View Documentation | Edit
TESS::TrainingSet		View Table | View Documentation | Edit
TESS::TrainingSetMember		View Table | View Documentation | Edit
Experiment		
Protocols		Return to Super Category
PROT::Action	PROT::Parameterizable	View Table | View Documentation | Edit
PROT::ActionInstance	PROT::ParameterizableInstance	View Table | View Documentation | Edit

TESS::FootprintMethodProtocol		View Table | View Documentation | Edit
PROT::Hardware	PROT::Parameterizable	View Table | View Documentation | Edit
RAD::LabelMethod		View Table | View Documentation | Edit
PROT::Parameter		View Table | View Documentation | Edit
PROT::ParameterInstance		View Table | View Documentation | Edit
PROT::Parameterizable		View Table | View Documentation | Edit
PROT::ParameterizableInstance		View Table | View Documentation | Edit
PROT::ParameterizableLink		View Table | View Documentation | Edit
PROT::Protocol	PROT::Parameterizable	View Table | View Documentation | Edit
PROT::ProtocolAction	PROT::ParameterizableLink	View Table | View Documentation | Edit
PROT::ProtocolHardware	PROT::ParameterizableLink	View Table | View Documentation | Edit
PROT::ProtocolInstance	PROT::ParameterizableInstance	View Table | View Documentation | Edit
RAD::ProtocolParam		View Table | View Documentation | Edit
RAD::ProtocolQCParam		View Table | View Documentation | Edit
PROT::ProtocolSoftware	PROT::ParameterizableLink	View Table | View Documentation | Edit
RAD::ProtocolStep		View Table | View Documentation | Edit
PROT::Software	PROT::Parameterizable	View Table | View Documentation | Edit
Data Transformations and Analyses		Return to Super Category
TESS::Analysis		View Table | View Documentation | Edit
RAD::AnalysisInput		View Table | View Documentation | Edit
RAD::AnalysisParam		View Table | View Documentation | Edit
RAD::AnalysisQCParam		View Table | View Documentation | Edit
RAD::AnalysisResult		View Table | View Documentation | Edit
RAD::ArrayStatTwoConditions	RAD::AnalysisResult	View Table | View Documentation | Edit
RAD::AssayAnalysis		View Table | View Documentation | Edit
PROT::DataTransformation		View Table | View Documentation | Edit
RAD::DataTransformationResult	RAD::AnalysisResult	View Table | View Documentation | Edit
RAD::HQSpecificity	RAD::AnalysisResult	View Table | View Documentation | Edit
RAD::LogicalGroup		View Table | View Documentation | Edit
RAD::LogicalGroupLink		View Table | View Documentation | Edit
TESS::ModelResult		View Table | View Documentation | Edit
TESS::PredictionResult		View Table | View Documentation | Edit
RAD::SAM	RAD::AnalysisResult	View Table | View Documentation | Edit
RAD::SpliceArrayAnalysisResult	RAD::AnalysisResult	View Table | View Documentation | Edit
Learning Models		Return to Super Category
TESS::ParameterGroup		View Table | View Documentation | Edit
TESS::ParameterSubgroup		View Table | View Documentation | Edit
TESS::ParameterValue		View Table | View Documentation | Edit
TESS::ParserItem		View Table | View Documentation | Edit
TESS::ParserItemEdgeType		View Table | View Documentation | Edit
TESS::ParserItemLink		View Table | View Documentation | Edit
Model-based Regulatory Feature Prediction		Return to Super Category
Bounded Collection Grammars		Return to Super Category
TESS::SbcgAnnotationFilter		View Table | View Documentation | Edit
TESS::SbcgAnnotationFilterTerm		View Table | View Documentation | Edit
TESS::SbcgAnnotationGuide		View Table | View Documentation | Edit

TESS::SbcgComparisonType		View Table | View Documentation | Edit
TESS::SbcgNonterminal		View Table | View Documentation | Edit
TESS::SbcgPathTermRelationType		View Table | View Documentation | Edit
TESS::SbcgProduction		View Table | View Documentation | Edit
TESS::SbcgProductionType		View Table | View Documentation | Edit
TESS::SbcgRecogConsensusString	TESS::SbcgRecognition	View Table | View Documentation | Edit
TESS::SbcgRecogGap	TESS::SbcgRecognition	View Table | View Documentation | Edit
TESS::SbcgRecogMultiObs		View Table | View Documentation | Edit
TESS::SbcgRecognition		View Table | View Documentation | Edit
TESS::SbcgRecogPathExpression	TESS::SbcgRecognition	View Table | View Documentation | Edit
TESS::SbcgRecogPosition	TESS::SbcgRecognition	View Table | View Documentation | Edit
TESS::SbcgRecogString	TESS::SbcgRecognition	View Table | View Documentation | Edit
TESS::SbcgRecogWeightMatrix	TESS::SbcgRecognition	View Table | View Documentation | Edit
TESS::SbcgRhsTerm		View Table | View Documentation | Edit
TESS::SbcgStream		View Table | View Documentation | Edit
TESS::SbcgStreamParameter		View Table | View Documentation | Edit
Platform		Return to Super Category
RAD::ArrayDesign		View Table | View Documentation | Edit
RAD::ArrayDesignAnnotation		View Table | View Documentation | Edit
RAD::ArrayDesignGroup		View Table | View Documentation | Edit
RAD::ArrayDesignGroupLink		View Table | View Documentation | Edit
RAD::CompositeElement		View Table | View Documentation | Edit
RAD::CompositeElementAnnotation		View Table | View Documentation | Edit
RAD::Control		View Table | View Documentation | Edit
RAD::Element		View Table | View Documentation | Edit
RAD::ElementAnnotation		View Table | View Documentation | Edit
PROT::Feature		View Table | View Documentation | Edit
PROT::FeatureInput		View Table | View Documentation | Edit
PROT::FeatureSet		View Table | View Documentation | Edit
PROT::FSGAnnotation		View Table | View Documentation | Edit
PROT::MALDIPlate	PROT::FeatureSet	View Table | View Documentation | Edit
PROT::MALDISpot	PROT::Feature	View Table | View Documentation | Edit
RAD::SAGETag	RAD::CompositeElement	View Table | View Documentation | Edit
RAD::SAGETagMapping	RAD::Element	View Table | View Documentation | Edit
RAD::ShortOligo	RAD::Element	View Table | View Documentation | Edit
RAD::ShortOligoFamily	RAD::CompositeElement	View Table | View Documentation | Edit
RAD::Spot	RAD::Element	View Table | View Documentation | Edit
RAD::SpotFamily	RAD::CompositeElement	View Table | View Documentation | Edit
Assay		Return to Super Category
PROT::Acquisition		View Table | View Documentation | Edit
PROT::AcquisitionChain		View Table | View Documentation | Edit
PROT::AcquisitionChannel		View Table | View Documentation | Edit
RAD::AcquisitionParam		View Table | View Documentation | Edit
PROT::AcquisitionQuantification		View Table | View Documentation | Edit
PROT::Assay		View Table | View Documentation | Edit
PROT::AssayAnnotation		View Table | View Documentation | Edit

PROT::AssayChain		View Table | View Documentation | Edit
RAD::AssayControl		View Table | View Documentation | Edit
PROT::AssayInput		View Table | View Documentation | Edit
RAD::AssayParam		View Table | View Documentation | Edit
PROT::AssayParentFile		View Table | View Documentation | Edit
PROT::AssayProduct		View Table | View Documentation | Edit
PROT::MZRun	PROT::Assay	View Table | View Documentation | Edit
PROT::MZScan	PROT::Acquisition	View Table | View Documentation | Edit
PROT::ParentFile		View Table | View Documentation | Edit
PROT::Quantification		View Table | View Documentation | Edit
RAD::QuantificationParam		View Table | View Documentation | Edit
RAD::RelatedAcquisition		View Table | View Documentation | Edit
RAD::RelatedQuantification		View Table | View Documentation | Edit

Quantified Data
Return to Super Category

RAD::AffymetrixCEL	RAD::ElementResult	View Table | View Documentation | Edit
RAD::AffymetrixMAS4	RAD::CompositeElementResult	View Table | View Documentation | Edit
RAD::AffymetrixMAS5	RAD::CompositeElementResult	View Table | View Documentation | Edit
RAD::AgilentElementResult	RAD::ElementResult	View Table | View Documentation | Edit
RAD::ArrayVisionElementResult	RAD::ElementResult	View Table | View Documentation | Edit
RAD::CompositeElementResult		View Table | View Documentation | Edit
PROT::ElementResult		View Table | View Documentation | Edit
RAD::GEMToolsElementResult	RAD::ElementResult	View Table | View Documentation | Edit
RAD::GenePixElementResult	RAD::ElementResult	View Table | View Documentation | Edit
RAD::MOIDResult	RAD::CompositeElementResult	View Table | View Documentation | Edit
PROT::MZPeak	PROT::Quantification	View Table | View Documentation | Edit
PROT::MZPeakResult	PROT::ElementResult	View Table | View Documentation | Edit
RAD::ProbeProfiler	RAD::CompositeElementResult	View Table | View Documentation | Edit
RAD::QuantArrayElementResult	RAD::ElementResult	View Table | View Documentation | Edit
RAD::RMAExpress	RAD::CompositeElementResult	View Table | View Documentation | Edit
RAD::SAGETagResult	RAD::CompositeElementResult	View Table | View Documentation | Edit
RAD::ScanAlyzeElementResult	RAD::ElementResult	View Table | View Documentation | Edit
RAD::SpotElementResult	RAD::ElementResult	View Table | View Documentation | Edit

Experimental Design
Return to Super Category

Study::Study		View Table | View Documentation | Edit
PROT::StudyAssay		View Table | View Documentation | Edit
PROT::StudyBioMaterial		View Table | View Documentation | Edit
Study::StudyDesign		View Table | View Documentation | Edit
PROT::StudyDesignAssay		View Table | View Documentation | Edit
Study::StudyDesignDescription		View Table | View Documentation | Edit
Study::StudyDesignType		View Table | View Documentation | Edit
Study::StudyFactor		View Table | View Documentation | Edit
PROT::StudyFactorValue		View Table | View Documentation | Edit

Biomaterials
Return to Super Category

RAD::AssayBioMaterial		View Table | View Documentation | Edit
RAD::AssayLabeledExtract		View Table | View Documentation | Edit
Study::BioMaterialCharacteristic		View Table | View Documentation | Edit

PROT::BioMaterialMeasurement		View Table | View Documentation | Edit
Study::BioSample	Study::BioMaterial	View Table | View Documentation | Edit
Study::BioSource	Study::BioMaterial	View Table | View Documentation | Edit
Study::BMCRelationship		View Table | View Documentation | Edit
Study::LabeledExtract	Study::BioMaterial	View Table | View Documentation | Edit
RAD::Treatment		View Table | View Documentation | Edit
RAD::TreatmentParam		View Table | View Documentation | Edit
(Mixed) Ontologies		Return to Super Category
Integrity		Return to Super Category
RAD::IntegrityStatInput		View Table | View Documentation | Edit
RAD::IntegrityStatistic		View Table | View Documentation | Edit

Provenance

Algorithm		Return to Super Category
Core::Algorithm		View Table | View Documentation | Edit
Core::AlgorithmImplementation		View Table | View Documentation | Edit
Core::AlgorithmInvocation		View Table | View Documentation | Edit
Core::AlgorithmParam		View Table | View Documentation | Edit
Core::AlgorithmParamKey		View Table | View Documentation | Edit
Core::AlgorithmParamKeyType		View Table | View Documentation | Edit
External Database		Return to Super Category
SRes::DbRef		View Table | View Documentation | Edit
SRes::ExternalDatabase		View Table | View Documentation | Edit
SRes::ExternalDatabaseEntry		View Table | View Documentation | Edit
SRes::ExternalDatabaseLink		View Table | View Documentation | Edit
SRes::ExternalDatabaseRelease		View Table | View Documentation | Edit
Bibliographic		Return to Super Category
SRes::Abstract		View Table | View Documentation | Edit
DoTS::Attribution		View Table | View Documentation | Edit
SRes::Author		View Table | View Documentation | Edit
SRes::BibliographicReference		View Table | View Documentation | Edit
SRes::BibRefAuthor		View Table | View Documentation | Edit
SRes::BibRefType		View Table | View Documentation | Edit
SRes::Contact		View Table | View Documentation | Edit
SRes::Reference		View Table | View Documentation | Edit
SRes::SequenceReference		View Table | View Documentation | Edit
Evidence		Return to Super Category
DoTS::Comments		View Table | View Documentation | Edit
DoTS::Evidence		View Table | View Documentation | Edit
SRes::ReviewStatus		View Table | View Documentation | Edit

Vocabularies

Anatomy Vocabulary		Return to Super Category
SRes::Anatomy		View Table | View Documentation | Edit
SRes::AnatomyLineage		View Table | View Documentation | Edit
DoTS::IsExpressed		View Table | View Documentation | Edit
SRes::Lineage		View Table | View Documentation | Edit
Gene Ontology		Return to Super Category

SRes::GOEvidenceCode		View Table | View Documentation | Edit
SRes::GORelationship		View Table | View Documentation | Edit
SRes::GORelationshipType		View Table | View Documentation | Edit
SRes::GOSynonym		View Table | View Documentation | Edit
SRes::GOTerm		View Table | View Documentation | Edit
Enzymes Vocabulary		Return to Super Category
DoTS::AASequenceEnzymeClass		View Table | View Documentation | Edit
SRes::ECPathway		View Table | View Documentation | Edit
SRes::ECPathwayEnzymeClass		View Table | View Documentation | Edit
SRes::EnzymeClass		View Table | View Documentation | Edit
SRes::EnzymeClassAttribute		View Table | View Documentation | Edit
Taxon Vocabulary		Return to Super Category
SRes::Taxon		View Table | View Documentation | Edit
SRes::TaxonName		View Table | View Documentation | Edit
Developmental Stage Vocabulary		Return to Super Category
SRes::DevelopmentalStage		View Table | View Documentation | Edit
Disease Vocabulary		Return to Super Category
SRes::Disease		View Table | View Documentation | Edit
Genetic Code Vocabulary		Return to Super Category
SRes::GeneticCode		View Table | View Documentation | Edit
MGED Ontology		Return to Super Category
Mutagens Vocabulary		Return to Super Category
SRes::Mutagen		View Table | View Documentation | Edit
Phenotype Vocabulary		Return to Super Category
SRes::PATOAttribute		View Table | View Documentation | Edit
SRes::Phenotype		View Table | View Documentation | Edit
SRes::PhenotypeClass		View Table | View Documentation | Edit
Sequence Ontology		Return to Super Category
SRes::SequenceOntology		View Table | View Documentation | Edit
Repeat Types Vocabulary		Return to Super Category
DoTS::RepeatType		View Table | View Documentation | Edit
Sequence Types Vocabulary		Return to Super Category
DoTS::SequenceType		View Table | View Documentation | Edit
Administration		
Meta Info		Return to Super Category
Core::DatabaseDocumentation		View Table | View Documentation | Edit
Core::DatabaseInfo		View Table | View Documentation | Edit
Core::TableCategory		View Table | View Documentation | Edit
Core::TableInfo		View Table | View Documentation | Edit
Administration		Return to Super Category
Core::GroupInfo		View Table | View Documentation | Edit
Core::ProjectInfo		View Table | View Documentation | Edit
RAD::ProjectLink		View Table | View Documentation | Edit
Core::UserGroup		View Table | View Documentation | Edit
Core::UserInfo		View Table | View Documentation | Edit
Core::UserProject		View Table | View Documentation | Edit

gusdb.org | Schema Browser

Misc Applications		Return to Super Category		
DoTS::AAMotifGOTermRule		View Table	View Documentation	Edit
DoTS::GeneFeatureSeqOverlap		View Table	View Documentation	Edit
Text Search		Return to Super Category		
DoTS::IndexWord		View Table	View Documentation	Edit
DoTS::IndexWordLink		View Table	View Documentation	Edit

webmaster@gusdb.org

GUS Schema >> DoTS::Assembly (subclass of DoTS::NASequence)

The Assembly view of NASequenceImp stores information and consensus sequences generated by assembling EST and mRNA sequences using the cap4 algorithm (Paracel).

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_SEQUENCE_ID	no	NUMBER(10,0)	Primary Key Edit
SEQUENCE_VERSION	no	NUMBER(3,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
SEQUENCE_TYPE_ID		DoTS::SequenceType (NUMBER(4,0))	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
TAXON_ID		SRes::Taxon (NUMBER(12,0))	Edit
SEQUENCE		CLOB	Edit
LENGTH		NUMBER(12,0)	Edit
A_COUNT		NUMBER(12,0)	Edit
C_COUNT		NUMBER(12,0)	Edit
G_COUNT		NUMBER(12,0)	Edit
T_COUNT		NUMBER(12,0)	Edit
OTHER_COUNT		NUMBER(12,0)	Edit
DESCRIPTION		STRING(2000)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SEQUENCE_PIECE_ID		DoTS::SequencePiece (NUMBER(10,0))	Edit
SEQUENCING_CENTER_CONTACT_ID		SRes::Contact (NUMBER(12,0))	Edit
FULL_LENGTH_CDS		NUMBER(3,0)	Edit
ASSEMBLY_CONSISTENCY		NUMBER(10,0)	Consistency of the assembly by scoring each column where consistency = number of majority nucleotide / total number in columnnormalized over length of assembly where depth > 1 Edit
CONTAINS_MRNA		NUMBER(1,0)	set to 1 if >= 1 contained AssemblySequence is of sequence type "mRNA" Edit
NUMBER_OF_CONTAINED_SEQUENCES		NUMBER(12,0)	The number of sequences contained in the Assembly Edit
NOTES		STRING(255)	Edit
GAPPED_CONSENSUS		CLOB	consensus sequence including the gaps inserted by cap4 Edit
QUALITY_VALUES		CLOB	string of quality values assigned by cap4 Edit

Child tables:

Subclasses:

GUS Schema >> DoTS::NASequence

column	nulls?	type	description
NA_SEQUENCE_ID	no	NUMBER(10,0)	Primary Key Edit
SEQUENCE_VERSION	no	NUMBER(3,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
SEQUENCE_TYPE_ID		DoTS::SequenceType (NUMBER(4,0))	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
TAXON_ID		SRes::Taxon (NUMBER(12,0))	Edit
SEQUENCE		CLOB	Edit
LENGTH		NUMBER(12,0)	Edit
A_COUNT		NUMBER(12,0)	Edit
C_COUNT		NUMBER(12,0)	Edit
G_COUNT		NUMBER(12,0)	Edit
T_COUNT		NUMBER(12,0)	Edit
OTHER_COUNT		NUMBER(12,0)	Edit
DESCRIPTION		STRING(2000)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SEQUENCE_PIECE_ID		DoTS::SequencePiece (NUMBER(10,0))	Edit
SEQUENCING_CENTER_CONTACT_ID		SRes::Contact (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

NA Sequence
DoTS::DbRefNASequence
DoTS::NASequence
DoTS::SequencePiece
NA Sequence Features
DoTS::NAFeature
Feature Relations
DoTS::ExonLocation
DoTS::GeneFeatureSAGETagLink
GenBank Sequence Record
DoTS::NAComment
DoTS::NAEntry
DoTS::NASeqCytoLocation
DoTS::NASequenceKeyword
DoTS::NASequenceOrganelle
DoTS::NASequenceRef
Similarity
DoTS::BLATAlignment
DoTS::ConsistentAlignment
Assembly
DoTS::AssemblyAnatomyPercent

    [DoTS::AssemblySequence](#)
    [DoTS::AssemblySNP](#)
    [DoTS::GeneTrapAssembly](#)
Mapping
    [DoTS::EndSequencePairMap](#)
    [DoTS::EPCR](#)
    [DoTS::OpticalMapAlignment](#)
Clones
    [DoTS::EST](#)
Repeat Types Vocabulary
    [DoTS::RepeatType](#)
Misc Applications
    [DoTS::GeneFeatureSeqOverlap](#)

Subclasses:

[DoTS::VirtualSequence](#)
[DoTS::ExternalNASequence](#)
[DoTS::SplicedNASequence](#)
[DoTS::Assembly](#)

GUS Schema >> DoTS::DbRefNASequence

Links DbRef to NASequence entries

column	nulls?	type	description
DB_REF_NA_SEQUENCE_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(10,0))	Edit
DB_REF_ID	no	SRes::DbRef (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

GUS Schema >> DoTS::ExternalNASequence (subclass of DoTS::NASequence)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_SEQUENCE_ID	no	NUMBER(10,0)	Primary Key Edit
SEQUENCE_VERSION	no	NUMBER(3,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
SEQUENCE_TYPE_ID		DoTS::SequenceType (NUMBER(4,0))	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
TAXON_ID		SRes::Taxon (NUMBER(12,0))	Edit
SEQUENCE		CLOB	Edit
LENGTH		NUMBER(12,0)	Edit
A_COUNT		NUMBER(12,0)	Edit
C_COUNT		NUMBER(12,0)	Edit
G_COUNT		NUMBER(12,0)	Edit
T_COUNT		NUMBER(12,0)	Edit
OTHER_COUNT		NUMBER(12,0)	Edit
DESCRIPTION		STRING(2000)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SEQUENCE_PIECE_ID		DoTS::SequencePiece (NUMBER(10,0))	Edit
SEQUENCING_CENTER_CONTACT_ID		SRes::Contact (NUMBER(12,0))	Edit
SOURCE_ID		STRING(255)	Edit
SECONDARY_IDENTIFIER		STRING(255)	Edit
NAME		STRING(255)	Edit
CHROMOSOME		STRING(255)	Edit
CHROMOSOME_ORDER_NUM		NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> DoTS::SequencePiece

column	nulls?	type	description
SEQUENCE_PIECE_ID	no	NUMBER(10,0)	[Edit](#)
VIRTUAL_NA_SEQUENCE_ID	no	[DoTS::NASequence](#) (NUMBER(10,0))	[Edit](#)
PIECE_NA_SEQUENCE_ID	no	[DoTS::NASequence](#) (NUMBER(10,0))	[Edit](#)
SEQUENCE_ORDER	no	NUMBER(5,0)	[Edit](#)
DISTANCE_FROM_LEFT		NUMBER(12,0)	[Edit](#)
UNCERTAINTY		FLOAT	[Edit](#)
STRAND_ORIENTATION		STRING(7)	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

NA Sequence
[DoTS::NASequence](#)

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::SplicedNASequence (subclass of DoTS::NASequence)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_SEQUENCE_ID	no	NUMBER(10,0)	Primary Key Edit
SEQUENCE_VERSION	no	NUMBER(3,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
SEQUENCE_TYPE_ID		DoTS::SequenceType (NUMBER(4,0))	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
TAXON_ID		SRes::Taxon (NUMBER(12,0))	Edit
SEQUENCE		CLOB	Edit
LENGTH		NUMBER(12,0)	Edit
A_COUNT		NUMBER(12,0)	Edit
C_COUNT		NUMBER(12,0)	Edit
G_COUNT		NUMBER(12,0)	Edit
T_COUNT		NUMBER(12,0)	Edit
OTHER_COUNT		NUMBER(12,0)	Edit
DESCRIPTION		STRING(2000)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SEQUENCE_PIECE_ID		DoTS::SequencePiece (NUMBER(10,0))	Edit
SEQUENCING_CENTER_CONTACT_ID		SRes::Contact (NUMBER(12,0))	Edit
SOURCE_ID		STRING(255)	Edit
SECONDARY_IDENTIFIER		STRING(255)	Edit
NAME		STRING(255)	Edit

Child tables:

Subclasses:

GUS Schema >> DoTS::VirtualSequence (subclass of DoTS::NASequence)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_SEQUENCE_ID	no	NUMBER(10,0)	Primary Key Edit
SEQUENCE_VERSION	no	NUMBER(3,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
SEQUENCE_TYPE_ID		DoTS::SequenceType (NUMBER(4,0))	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
TAXON_ID		SRes::Taxon (NUMBER(12,0))	Edit
SEQUENCE		CLOB	Edit
LENGTH		NUMBER(12,0)	Edit
A_COUNT		NUMBER(12,0)	Edit
C_COUNT		NUMBER(12,0)	Edit
G_COUNT		NUMBER(12,0)	Edit
T_COUNT		NUMBER(12,0)	Edit
OTHER_COUNT		NUMBER(12,0)	Edit
DESCRIPTION		STRING(2000)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SEQUENCE_PIECE_ID		DoTS::SequencePiece (NUMBER(10,0))	Edit
SEQUENCING_CENTER_CONTACT_ID		SRes::Contact (NUMBER(12,0))	Edit
CONFIDENCE		STRING(255)	Edit
SOURCE_ID		STRING(255)	Edit
SECONDARY_IDENTIFIER		STRING(255)	Edit
CHROMOSOME		STRING(255)	Edit
CHROMOSOME_ORDER_NUM		NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::AASequence

column	nulls?	type	description
AA_SEQUENCE_ID	no	NUMBER(10,0)	Edit
SEQUENCE_VERSION		NUMBER(3,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
MOLECULAR_WEIGHT		NUMBER(12,0)	Edit
SEQUENCE		CLOB	Edit
LENGTH		NUMBER(12,0)	Edit
DESCRIPTION		STRING(2000)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(32)	Edit
SOURCE_AA_SEQUENCE_ID		DoTS::AASequence (NUMBER(10,0))	Edit
SEQUENCE_TYPE_ID		DoTS::SequenceType (NUMBER(4,0))	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
TAXON_ID		SRes::Taxon (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

AA Sequence
    DoTS::AASequence
    DoTS::NRDBEntry
AA Sequence Features
    DoTS::AAFeature
GenBank Sequence Record
    DoTS::AAComment
    DoTS::AAEntry
    DoTS::AAGene
    DoTS::AASequenceKeyword
    DoTS::AASequenceOrganelle
    DoTS::AASequenceRef
    DoTS::AASequenceTaxon
Central Dogma
    DoTS::ProteinProperty
AA Ortholog
    DoTS::AASequenceSequenceGroup
Enzymes Vocabulary
    DoTS::AASequenceEnzymeClass
Uncategorized
    DoTS::AAMotifGOTermRuleSet
    DoTS::AASequenceDbRef
    DoTS::SecondaryStructure

Subclasses:

DoTS::MotifAASequence

DoTS::ExternalAASequence
DoTS::TrivialTranslation
DoTS::TranslatedAASequence

GUS Schema >> DoTS::DbRefPfamEntry

column	nulls?	type	description
DB_REF_PFAM_ENTRY_ID	no	NUMBER(10,0)	Edit
PFAM_ENTRY_ID	no	DoTS::PfamEntry (NUMBER(10,0))	Edit
DB_REF_ID	no	SRes::DbRef (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

GUS Schema >> DoTS::ExternalAASequence (subclass of DoTS::AASequence)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_SEQUENCE_ID	no	NUMBER(10,0)	Edit
SEQUENCE_VERSION		NUMBER(3,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
MOLECULAR_WEIGHT		NUMBER(12,0)	Edit
SEQUENCE		CLOB	Edit
LENGTH		NUMBER(12,0)	Edit
DESCRIPTION		STRING(2000)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(32)	Edit
SOURCE_AA_SEQUENCE_ID		DoTS::AASequence (NUMBER(10,0))	Edit
SEQUENCE_TYPE_ID		DoTS::SequenceType (NUMBER(4,0))	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
TAXON_ID		SRes::Taxon (NUMBER(10,0))	Edit
SECONDARY_IDENTIFIER		STRING(255)	Edit
NAME		STRING(255)	Edit
MOLECULE_TYPE		STRING(255)	Edit
CRC32_VALUE		STRING(255)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::MotifAASequence (subclass of DoTS::AASequence)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_SEQUENCE_ID	no	NUMBER(10,0)	Edit
SEQUENCE_VERSION		NUMBER(3,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
MOLECULAR_WEIGHT		NUMBER(12,0)	Edit
SEQUENCE		CLOB	Edit
LENGTH		NUMBER(12,0)	Edit
DESCRIPTION		STRING(2000)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(32)	Edit
SOURCE_AA_SEQUENCE_ID		DoTS::AASequence (NUMBER(10,0))	Edit
SEQUENCE_TYPE_ID		DoTS::SequenceType (NUMBER(4,0))	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
TAXON_ID		SRes::Taxon (NUMBER(10,0))	Edit
NUMBER_OF_CONTAINED_SEQUENCES		NUMBER(12,0)	Edit
SECONDARY_IDENTIFIER		STRING(255)	Edit
NAME		STRING(255)	Edit
MOLECULE_TYPE		STRING(255)	Edit
CRC32_VALUE		STRING(255)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::NRDBEntry

column	nulls?	type	description
NRDB_ENTRY_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
GID		STRING(255)	Edit
SOURCE_ID	no	STRING(32)	Edit
SEQUENCE_VERSION		NUMBER(3,0)	Edit
EXTERNAL_DATABASE_RELEASE_ID	no	SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
DESCRIPTION		STRING(255)	Edit
TAXON_ID		SRes::Taxon (NUMBER(12,0))	Edit
IS_PREFERRED	no	NUMBER(1,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

[GUS Schema](#) >> DoTS::PfamEntry

column	nulls?	type	description
PFAM_ENTRY_ID	no	NUMBER(10,0)	Edit
RELEASE	no	STRING(5)	Edit
ACCESSION	no	STRING(10)	Edit
IDENTIFIER	no	STRING(32)	Edit
DEFINITION	no	STRING(80)	Edit
AUTHOR		STRING(255)	Edit
ALIGNMENT_METHOD		STRING(100)	Edit
NUMBER_OF_SEQS	no	NUMBER(12,0)	Edit
COMMENT_STRING		CLOB	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

AA Sequence
   DoTS::DbRefPfamEntry
AA Sequence Features
   DoTS::AAFeature

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::TranslatedAASequence (subclass of DoTS::AASequence)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_SEQUENCE_ID	no	NUMBER(10,0)	Edit
SEQUENCE_VERSION		NUMBER(3,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
MOLECULAR_WEIGHT		NUMBER(12,0)	Edit
SEQUENCE		CLOB	Edit
LENGTH		NUMBER(12,0)	Edit
DESCRIPTION		STRING(2000)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(32)	Edit
SOURCE_AA_SEQUENCE_ID		DoTS::AASequence (NUMBER(10,0))	Edit
SEQUENCE_TYPE_ID		DoTS::SequenceType (NUMBER(4,0))	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
TAXON_ID		SRes::Taxon (NUMBER(10,0))	Edit
NOTES		STRING(255)	Edit
IS_SIMPLE		NUMBER(12,0)	Edit
SECONDARY_IDENTIFIER		STRING(255)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::TrivialTranslation (subclass of DoTS::AASequence)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_SEQUENCE_ID	no	NUMBER(10,0)	Edit
SEQUENCE_VERSION		NUMBER(3,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
MOLECULAR_WEIGHT		NUMBER(12,0)	Edit
SEQUENCE		CLOB	Edit
LENGTH		NUMBER(12,0)	Edit
DESCRIPTION		STRING(2000)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(32)	Edit
SOURCE_AA_SEQUENCE_ID		DoTS::AASequence (NUMBER(10,0))	Edit
SEQUENCE_TYPE_ID		DoTS::SequenceType (NUMBER(4,0))	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
TAXON_ID		SRes::Taxon (NUMBER(10,0))	Edit
READING_FRAME		NUMBER(3,0)	Edit
CODON_TABLE		STRING(255)	Edit
TRANSLATION_START		NUMBER(12,0)	Edit
TRANSLATION_STOP		NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::AlleleFeature (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
DESCRIPTION		STRING(4000)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

gusdb.org | Schema Browser

GUS Schema >> DoTS::NAFeature

Superclass view from NAFeature views

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

NA Sequence Features
DoTS::NAFeature
AA Sequence Features
DoTS::AAFeature
Feature Relations
DoTS::DbRefNAFeature
DoTS::GeneFeatureSAGETagLink
DoTS::NAFeatureComment
DoTS::NAFeatureRelationship
DoTS::NALocation
DoTS::RNAFeatureExon
GenBank Sequence Record
DoTS::NAFeatureNAGene
DoTS::NAFeatureNAProtein
DoTS::NAFeatureNAPT
Central Dogma
DoTS::GeneInstance
DoTS::RNAInstance
Mapping
DoTS::OpticalMapAlignmentSpan
Variation
DoTS::AlleleInstance
High Volume Genome Annotation
TESS::RegionOfInterest
Training Sets
TESS::Footprint
Learning Models

TESS::ParserItem
Misc Applications
DoTS::GeneFeatureSeqOverlap
Uncategorized
DoTS::PhylogeneticProfile

Subclasses:

DoTS::DNAStructure
DoTS::IntronFeature
DoTS::Source
DoTS::BindingSiteFeature
DoTS::SeqVariation
DoTS::TandemRepeatFeature
DoTS::RNAType
DoTS::AlleleFeature
DoTS::RNAFeature
DoTS::RNAStructure
DoTS::BindingSiteFeatureVer
DoTS::TransposableElement
DoTS::CollectionNAFeature
DoTS::SAGETagFeature
DoTS::ReplicationOriginFeature
DoTS::ProteinFeature
DoTS::HexamerFeature
DoTS::UTRFeature
DoTS::RNASecondaryStructure
DoTS::ExonFeature
DoTS::ScaffoldGapFeature
DoTS::RestrictionFragmentFeature
DoTS::RNARegulatoryFeature
DoTS::InflectionPointFeature
DoTS::PromoterFeature
DoTS::ChromosomeElementFeature
DoTS::PolyAFeature
DoTS::LowComplexityNAFeature
DoTS::Miscellaneous
DoTS::SpliceSiteFeature
DoTS::GeneFeature
DoTS::Immunoglobulin
DoTS::STS
DoTS::DNARegulatory
DoTS::Repeats
DoTS::Transcript

webmaster@gusdb.org

GUS Schema >> DoTS::BindingSiteFeature (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
MODEL		STRING(255)	Edit
MODEL_ID		NUMBER(12,0)	Edit
PRIMARY_SCORE		FLOAT	Edit
SECONDARY_SCORE		FLOAT	Edit
SYNDROME		STRING(1000)	Edit

Child tables:

Subclasses:

GUS Schema >> DoTS::ChromosomeElementFeature (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
DESCRIPTION		STRING(1000)	Edit
LENGTH		NUMBER(12,0)	Edit
CHROMOSOME_END		STRING(255)	Edit
PERCENT_AT		FLOAT	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> **DoTS::CollectionNAFeature** (subclass of [DoTS::NAFeature](#))

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		[DoTS::NASequence](#) (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		[SRes::SequenceOntology](#) (NUMBER(10,0))	Edit
PARENT_ID		[DoTS::NAFeature](#) (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		[SRes::ExternalDatabaseRelease](#) (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		[Core::Algorithm](#) (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		[SRes::ReviewStatus](#) (NUMBER(10,0))	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

gusdb.org | Schema Browser

GUS Schema >> DoTS::DNARegulatory (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
CITATION		STRING(1000)	Edit
EVIDENCE		STRING(255)	Edit
FUNCTION		STRING(255)	Edit
GENE		STRING(255)	Edit
LABEL		STRING(2000)	Edit
MAP		STRING(255)	Edit
PARTIAL		STRING(255)	Edit
PHENOTYPE		STRING(255)	Edit
STANDARD_NAME		STRING(1000)	Edit
USEDIN		STRING(1024)	Edit
IS_PARTIAL		NUMBER(3,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::DNAStructure (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
CITATION		STRING(1000)	Edit
DIRECTION		STRING(255)	Edit
EVIDENCE		STRING(255)	Edit
FUNCTION		STRING(255)	Edit
GENE		STRING(2000)	Edit
LABEL		STRING(255)	Edit
MAP		STRING(255)	Edit
PARTIAL		STRING(255)	Edit
PCR_CONDITIONS		STRING(1000)	Edit
STANDARD_NAME		STRING(1024)	Edit
USEDIN		STRING(255)	Edit
IS_PARTIAL		NUMBER(3,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::ExonFeature (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
IS_ONE_OF		NUMBER(3,0)	Edit
IS_OPTIONAL		NUMBER(3,0)	Edit
ORDER_NUMBER		NUMBER(12,0)	Edit
VERSION		STRING(1000)	Edit
NUMBER_OF_ALTERNATES		NUMBER(3,0)	Edit
IS_INITIAL_EXON		NUMBER(3,0)	Edit
IS_FINAL_EXON		NUMBER(3,0)	Edit
IS_MERGED_EXON		NUMBER(12,0)	Edit
CODING_START		NUMBER(12,0)	Edit
CODING_END		NUMBER(12,0)	Edit
READING_FRAME		NUMBER(3,0)	Edit
SCORE		FLOAT	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::GeneFeature (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
GENE_TYPE		STRING(500)	Edit
CONFIRMED_BY_SIMILARITY		NUMBER(3,0)	Edit
PREDICTION_NUMBER		NUMBER(12,0)	Edit
NUMBER_OF_EXONS		NUMBER(12,0)	Edit
HAS_INITIAL_EXON		NUMBER(3,0)	Edit
HAS_FINAL_EXON		NUMBER(3,0)	Edit
SCORE		FLOAT	Edit
SECONDARY_SCORE		FLOAT	Edit
IS_PSEUDO		NUMBER(3,0)	Edit
IS_PARTIAL		NUMBER(3,0)	Edit
ALLELE		STRING(1000)	Edit
CITATION		STRING(255)	Edit
EVIDENCE		STRING(255)	Edit
FUNCTION		STRING(255)	Edit
GENE		STRING(2000)	Edit
LABEL		STRING(255)	Edit
MAP		STRING(255)	Edit
PHENOTYPE		STRING(1000)	Edit
PRODUCT		STRING(1024)	Edit
STANDARD_NAME		STRING(255)	Edit
USEDIN		STRING(1000)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::HexamerFeature (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
SCORE		FLOAT	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::Immunoglobulin (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
CITATION		STRING(1000)	Edit
CODON		STRING(255)	Edit
CODON_START		NUMBER(12,0)	Edit
EVIDENCE		STRING(255)	Edit
GENE		STRING(255)	Edit
LABEL		STRING(255)	Edit
MAP		STRING(2000)	Edit
PARTIAL		STRING(255)	Edit
PRODUCT		STRING(255)	Edit
PSEUDO		STRING(1000)	Edit
TRANSL_EXCEPT		STRING(1024)	Edit
TRANSLATION		CLOB	Edit
TRANSL_TABLE		NUMBER(12,0)	Edit
STANDARD_NAME		STRING(255)	Edit
USEDIN		STRING(255)	Edit
IS_PARTIAL		NUMBER(3,0)	Edit
IS_PSEUDO		NUMBER(3,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::InflectionPointFeature (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
DESCRIPTION		STRING(1000)	Edit

Child tables:

Subclasses:

GUS Schema >> DoTS::IntronFeature (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
IS_ONE_OF		NUMBER(3,0)	Edit
ORDER_NUMBER		NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::LowComplexityNAFeature (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
LENGTH		NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::Miscellaneous (subclass of DoTS::NAFeature)

Miscellaneous GenBank features go here

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
BOUND_MOIETY		STRING(1000)	GenBank qualifier Edit
CITATION		STRING(255)	GenBank qualifier Edit
EVIDENCE		STRING(255)	GenBank qualifier Edit
FUNCTION		STRING(255)	GenBank qualifier Edit
GENE		STRING(2000)	GenBank qualifier Edit
LABEL		STRING(255)	GenBank qualifier Edit
MAP		STRING(255)	GenBank qualifier Edit
PARTIAL		STRING(255)	GenBank qualifier Edit
PCR_CONDITIONS		STRING(1000)	GenBank qualifier Edit
PHENOTYPE		STRING(1024)	GenBank qualifier Edit
PRODUCT		STRING(255)	GenBank qualifier Edit
PSEUDO		STRING(255)	GenBank qualifier Edit
STANDARD_NAME		STRING(1000)	GenBank qualifier Edit
USEDIN		STRING(500)	GenBank qualifier Edit
IS_PARTIAL		NUMBER(3,0)	Bit to say if this a partial sequence Edit
IS_PSEUDO		NUMBER(3,0)	Bit to say if this a pseudo sequence Edit
NUM		NUMBER(12,0)	GenBank qualifier Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::PolyAFeature (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
SCORE		FLOAT	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::PromoterFeature (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
SCORE		FLOAT	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

[GUS Schema]() >> DoTS::ProteinFeature (subclass of [DoTS::NAFeature]())

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		[DoTS::NASequence]() (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		[SRes::SequenceOntology]() (NUMBER(10,0))	Edit
PARENT_ID		[DoTS::NAFeature]() (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		[SRes::ExternalDatabaseRelease]() (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		[Core::Algorithm]() (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		[SRes::ReviewStatus]() (NUMBER(10,0))	Edit
CITATION		STRING(1000)	Edit
CLONE		STRING(255)	Edit
CODON		STRING(255)	Edit
CODON_START		NUMBER(12,0)	Edit
EC_NUMBER		STRING(255)	Edit
EVIDENCE		STRING(2000)	Edit
FUNCTION		STRING(255)	Edit
GENE		STRING(255)	Edit
LABEL		STRING(255)	Edit
MAP		STRING(1000)	Edit
PARTIAL		STRING(1024)	Edit
PRODUCT		STRING(255)	Edit
STANDARD_NAME		STRING(255)	Edit
TRANSL_EXCEPT		STRING(1000)	Edit
USEDIN		STRING(500)	Edit
IS_PARTIAL		NUMBER(3,0)	Edit

Child tables:

Subclasses:

[webmaster@gusdb.org]()

GUS Schema >> DoTS::Repeats (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
CITATION		STRING(1000)	Edit
EVIDENCE		STRING(255)	Edit
FUNCTION		STRING(255)	Edit
GENE		STRING(255)	Edit
LABEL		STRING(2000)	Edit
MAP		STRING(255)	Edit
PARTIAL		STRING(255)	Edit
RPT_FAMILY		STRING(255)	Edit
RPT_TYPE		STRING(1000)	Edit
RPT_UNIT		STRING(1024)	Edit
STANDARD_NAME		STRING(255)	Edit
USEDIN		STRING(255)	Edit
IS_PARTIAL		NUMBER(3,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::ReplicationOriginFeature (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
DESCRIPTION		STRING(1000)	Edit

Child tables:

Subclasses:

GUS Schema >> DoTS::RestrictionFragmentFeature (subclass of DoTS::NAFeature)

This view of NAFeatureImp is for restriction enzyme fragment.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
ENZYME_NAME		STRING(1000)	Edit
TYPE_OF_CUT		STRING(255)	The type of cut, sticky or blunt. Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::RNAFeature (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
NUMBER_OF_EXONS		NUMBER(12,0)	Edit
TRANSLATION_START		NUMBER(12,0)	Edit
TRANSLATION_STOP		NUMBER(12,0)	Edit

Child tables:

Subclasses:

GUS Schema >> DoTS::RNARegulatoryFeature (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
SCORE		FLOAT	Edit
FUNCTION		STRING(1000)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::RNAStructure (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
CITATION		STRING(1000)	Edit
EVIDENCE		STRING(255)	Edit
FUNCTION		STRING(255)	Edit
GENE		STRING(255)	Edit
LABEL		STRING(2000)	Edit
MAP		STRING(255)	Edit
PARTIAL		STRING(255)	Edit
PRODUCT		STRING(255)	Edit
STANDARD_NAME		STRING(1000)	Edit
USEDIN		STRING(1024)	Edit
IS_PARTIAL		NUMBER(3,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::RNAType (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
ANTICODON		STRING(1000)	Edit
CITATION		STRING(255)	Edit
CODON		STRING(255)	Edit
EVIDENCE		STRING(255)	Edit
FUNCTION		STRING(2000)	Edit
GENE		STRING(255)	Edit
LABEL		STRING(255)	Edit
MAP		STRING(255)	Edit
PARTIAL		STRING(1000)	Edit
PRODUCT		STRING(1024)	Edit
PSEUDO		STRING(255)	Edit
STANDARD_NAME		STRING(255)	Edit
USEDIN		STRING(1000)	Edit
IS_PARTIAL		NUMBER(3,0)	Edit
IS_PSEUDO		NUMBER(3,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::SAGETagFeature (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
RESTRICTION_ENZYME		STRING(1000)	Edit
BINDING_LOCATION_ID		NUMBER(12,0)	Edit
TRAILER_LOCATION_ID		NUMBER(12,0)	Edit
TAG_LOCATION_ID		NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::ScaffoldGapFeature (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
MIN_SIZE		NUMBER(12,0)	Edit
MAX_SIZE		NUMBER(12,0)	Edit
LEFT_CONTIG		STRING(255)	Edit
RIGHT_CONTIG		STRING(255)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::SeqVariation (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
CITATION		STRING(1000)	Edit
CLONE		STRING(255)	Edit
EVIDENCE		STRING(255)	Edit
FUNCTION		STRING(2000)	Edit
GENE		STRING(255)	Edit
LABEL		STRING(255)	Edit
MAP		STRING(255)	Edit
ORGANISM		STRING(1000)	Edit
STRAIN		STRING(255)	Edit
PARTIAL		STRING(1024)	Edit
PHENOTYPE		STRING(255)	Edit
PRODUCT		STRING(255)	Edit
STANDARD_NAME		STRING(1000)	Edit
SUBSTITUTE		STRING(4000)	Edit
NUM		STRING(255)	Edit
USEDIN		STRING(255)	Edit
MOD_BASE		STRING(255)	Edit
IS_PARTIAL		NUMBER(3,0)	Edit
FREQUENCY		FLOAT	Edit
ALLELE		STRING(255)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::Source (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
CELL_LINE		STRING(1000)	Edit
CELL_TYPE		STRING(255)	Edit
CHROMOPLAST		STRING(255)	Edit
CHROMOSOME		STRING(255)	Edit
CLONE		STRING(2000)	Edit
CLONE_LIB		STRING(255)	Edit
CULTIVAR		STRING(255)	Edit
CYANELLE		STRING(255)	Edit
DEV_STAGE		STRING(1000)	Edit
FOCUS		STRING(1024)	Edit
FREQUENCY		STRING(255)	Edit
GERMLINE		STRING(255)	Edit
HAPLOTYPE		STRING(1000)	Edit
INSERTION_SEQ		CLOB	Edit
ISOLATE		STRING(500)	Edit
KINETOPLAST		STRING(255)	Edit
LAB_HOST		STRING(255)	Edit
MACRONUCLEAR		STRING(255)	Edit
ORGANELLE		STRING(255)	Edit
POP_VARIANT		STRING(255)	Edit
PLASMID		STRING(4000)	Edit
PROVIRAL		STRING(255)	Edit
REARRANGED		STRING(255)	Edit
SEQUENCED_MOL		STRING(255)	Edit
SEROTYPE		STRING(255)	Edit
SEX		STRING(255)	Edit
SPECIFIC_HOST		STRING(255)	Edit
STRAIN		STRING(255)	Edit
SUB_CLONE		STRING(255)	Edit
SUB_SPECIES		STRING(255)	Edit
SUB_STRAIN		STRING(255)	Edit
TISSUE_LIB		STRING(255)	Edit
TRANSPOSON		STRING(255)	Edit
VARIETY		STRING(255)	Edit
VIRION		STRING(255)	Edit
CHLOROPLAST		STRING(255)	Edit
CITATION		STRING(255)	Edit
MAP		STRING(255)	Edit

ORGANISM	STRING(255)	Edit
SPECIMEN_VOUCHER	STRING(255)	Edit
TISSUE_TYPE	STRING(255)	Edit
USEDIN	STRING(255)	Edit
LABEL	STRING(255)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::SpliceSiteFeature (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
SCORE		FLOAT	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::STS (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
CITATION		STRING(1000)	Edit
EVIDENCE		STRING(255)	Edit
GENE		STRING(255)	Edit
LABEL		STRING(255)	Edit
MAP		STRING(2000)	Edit
PARTIAL		STRING(255)	Edit
STANDARD_NAME		STRING(255)	Edit
USEDIN		STRING(255)	Edit
IS_PARTIAL		NUMBER(3,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::TandemRepeatFeature (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
PERIOD		NUMBER(12,0)	Edit
COPYNUM		FLOAT	Edit
CONSENSUS_SIZE		NUMBER(12,0)	Edit
PERCENT_MATCH		NUMBER(3,0)	Edit
PERCENT_INDEL		NUMBER(3,0)	Edit
SCORE		FLOAT	Edit
A_COUNT		NUMBER(12,0)	Edit
C_COUNT		NUMBER(12,0)	Edit
G_COUNT		NUMBER(12,0)	Edit
T_COUNT		NUMBER(12,0)	Edit
ENTROPY		FLOAT	Edit
CONSENSUS		STRING(1000)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::Transcript (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
CITATION		STRING(1000)	Edit
CLONE		STRING(255)	Edit
CODON		STRING(255)	Edit
CODON_START		NUMBER(12,0)	Edit
CONS_SPLICE		STRING(255)	Edit
EC_NUMBER		STRING(2000)	Edit
EVIDENCE		STRING(255)	Edit
FUNCTION		STRING(255)	Edit
GENE		STRING(255)	Edit
LABEL		STRING(1000)	Edit
MAP		STRING(1024)	Edit
NUM		STRING(255)	Edit
PARTIAL		STRING(255)	Edit
PRODUCT		STRING(1000)	Edit
PROTEIN_ID		STRING(500)	Edit
PSEUDO		STRING(255)	Edit
STANDARD_NAME		STRING(255)	Edit
TRANSLATION		CLOB	Edit
TRANSL_EXCEPT		STRING(255)	Edit
TRANSL_TABLE		NUMBER(12,0)	Edit
USEDIN		STRING(255)	Edit
IS_PARTIAL		NUMBER(3,0)	Edit
IS_PSEUDO		NUMBER(3,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::TransposableElement (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::UTRFeature (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::AAFeature

column	nulls?	type	description
AA_Feature_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
Feature_NAME_ID		DoTS::FeatureName (NUMBER(10,0))	Edit
PARENT_ID		DoTS::AAFeature (NUMBER(10,0))	Edit
NA_Feature_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
DESCRIPTION		STRING(4000)	Edit
PFAM_ENTRY_ID		DoTS::PfamEntry (NUMBER(10,0))	Edit
MOTIF_AA_SEQUENCE_ID		DoTS::AASequence (NUMBER(10,0))	Edit
REPEAT_TYPE_ID		DoTS::RepeatType (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED	no	NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

AA Sequence Features
DoTS::AAFeature
DoTS::TranslatedAAFeatSeg
Feature Relations
DoTS::AALocation
Central Dogma
DoTS::ProteinInstance

Subclasses:

DoTS::PostTranslationalModFeature
DoTS::SignalPeptideFeature
DoTS::TertiaryStructureAAFeature
DoTS::RepeatRegionAAFeature
DoTS::LowComplexityAAFeature
DoTS::PlasmoAPFeature
DoTS::MassSpecFeature
DoTS::EpitopeFeature
DoTS::SecondaryStructureAAFeature
DoTS::TranslatedAAFeature
DoTS::PredictedAAFeature
DoTS::AASeqVariation
DoTS::DomainFeature

gusdb.org | Schema Browser

GUS Schema >> DoTS::AASeqVariation (subclass of DoTS::AAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_Feature_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
Feature_NAME_ID		DoTS::FeatureName (NUMBER(10,0))	Edit
PARENT_ID		DoTS::AAFeature (NUMBER(10,0))	Edit
NA_Feature_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
DESCRIPTION		STRING(4000)	Edit
PFAM_ENTRY_ID		DoTS::PfamEntry (NUMBER(10,0))	Edit
MOTIF_AA_SEQUENCE_ID		DoTS::AASequence (NUMBER(10,0))	Edit
REPEAT_TYPE_ID		DoTS::RepeatType (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED	no	NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
IS_SYNONYMOUS		NUMBER(3,0)	Edit
IS_STOP		NUMBER(3,0)	Edit
ORIGINAL		STRING(255)	Edit
SUBSTITUTE		STRING(255)	Edit
SPECIES		STRING(255)	Edit
STRAIN		STRING(255)	Edit
GENOMIC_LOCN		NUMBER(12,0)	Edit
NA_ORIGINAL		STRING(255)	Edit
NA_SUBSTITUTE		STRING(4000)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

[GUS Schema](#) >> DoTS::DomainFeature (subclass of [DoTS::AAFeature](#))

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_Feature_ID	no	NUMBER(10,0)	[Edit](#)
AA_SEQUENCE_ID	no	[DoTS::AASequence](#) (NUMBER(10,0))	[Edit](#)
Feature_NAME_ID		[DoTS::FeatureName](#) (NUMBER(10,0))	[Edit](#)
PARENT_ID		[DoTS::AAFeature](#) (NUMBER(10,0))	[Edit](#)
NA_Feature_ID		[DoTS::NAFeature](#) (NUMBER(10,0))	[Edit](#)
SUBCLASS_VIEW		STRING(30)	[Edit](#)
SEQUENCE_ONTOLOGY_ID		[SRes::SequenceOntology](#) (NUMBER(10,0))	[Edit](#)
DESCRIPTION		STRING(4000)	[Edit](#)
PFAM_ENTRY_ID		[DoTS::PfamEntry](#) (NUMBER(10,0))	[Edit](#)
MOTIF_AA_SEQUENCE_ID		[DoTS::AASequence](#) (NUMBER(10,0))	[Edit](#)
REPEAT_TYPE_ID		[DoTS::RepeatType](#) (NUMBER(10,0))	[Edit](#)
EXTERNAL_DATABASE_RELEASE_ID		[SRes::ExternalDatabaseRelease](#) (NUMBER(10,0))	[Edit](#)
SOURCE_ID		STRING(50)	[Edit](#)
PREDICTION_ALGORITHM_ID		[Core::Algorithm](#) (NUMBER(5,0))	[Edit](#)
IS_PREDICTED	no	NUMBER(1,0)	[Edit](#)
REVIEW_STATUS_ID		[SRes::ReviewStatus](#) (NUMBER(12,0))	[Edit](#)
NAME		STRING(255)	[Edit](#)
ALGORITHM_NAME		STRING(255)	[Edit](#)
SCORE		FLOAT	[Edit](#)
E_VALUE		FLOAT	[Edit](#)
PROBABILITY		FLOAT	[Edit](#)
NUMBER_OF_DOMAINS		NUMBER(12,0)	[Edit](#)

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::EpitopeFeature (subclass of DoTS::AAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_Feature_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
Feature_NAME_ID		DoTS::FeatureName (NUMBER(10,0))	Edit
PARENT_ID		DoTS::AAFeature (NUMBER(10,0))	Edit
NA_Feature_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
DESCRIPTION		STRING(4000)	Edit
PFAM_ENTRY_ID		DoTS::PfamEntry (NUMBER(10,0))	Edit
MOTIF_AA_SEQUENCE_ID		DoTS::AASequence (NUMBER(10,0))	Edit
REPEAT_TYPE_ID		DoTS::RepeatType (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED	no	NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
HAPLOTYPE		STRING(255)	Edit
TYPE		STRING(255)	Edit
SCORE		FLOAT	Edit
MAX_SCORE		FLOAT	Edit
WIDTH		NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::LowComplexityAAFeature (subclass of DoTS::AAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_Feature_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
Feature_NAME_ID		DoTS::FeatureName (NUMBER(10,0))	Edit
PARENT_ID		DoTS::AAFeature (NUMBER(10,0))	Edit
NA_Feature_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
DESCRIPTION		STRING(4000)	Edit
PFAM_ENTRY_ID		DoTS::PfamEntry (NUMBER(10,0))	Edit
MOTIF_AA_SEQUENCE_ID		DoTS::AASequence (NUMBER(10,0))	Edit
REPEAT_TYPE_ID		DoTS::RepeatType (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED	no	NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::MassSpecFeature (subclass of DoTS::AAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_Feature_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
Feature_NAME_ID		DoTS::FeatureName (NUMBER(10,0))	Edit
PARENT_ID		DoTS::AAFeature (NUMBER(10,0))	Edit
NA_Feature_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
DESCRIPTION		STRING(4000)	Edit
PFAM_ENTRY_ID		DoTS::PfamEntry (NUMBER(10,0))	Edit
MOTIF_AA_SEQUENCE_ID		DoTS::AASequence (NUMBER(10,0))	Edit
REPEAT_TYPE_ID		DoTS::RepeatType (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED	no	NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
DEVELOPMENTAL_STAGE		STRING(255)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::PlasmoAPFeature (subclass of DoTS::AAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_Feature_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
Feature_NAME_ID		DoTS::FeatureName (NUMBER(10,0))	Edit
PARENT_ID		DoTS::AAFeature (NUMBER(10,0))	Edit
NA_Feature_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
DESCRIPTION		STRING(4000)	Edit
PFAM_ENTRY_ID		DoTS::PfamEntry (NUMBER(10,0))	Edit
MOTIF_AA_SEQUENCE_ID		DoTS::AASequence (NUMBER(10,0))	Edit
REPEAT_TYPE_ID		DoTS::RepeatType (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED	no	NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::PostTranslationalModFeature (subclass of DoTS::AAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_Feature_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
Feature_NAME_ID		DoTS::FeatureName (NUMBER(10,0))	Edit
PARENT_ID		DoTS::AAFeature (NUMBER(10,0))	Edit
NA_Feature_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
DESCRIPTION		STRING(4000)	Edit
PFAM_ENTRY_ID		DoTS::PfamEntry (NUMBER(10,0))	Edit
MOTIF_AA_SEQUENCE_ID		DoTS::AASequence (NUMBER(10,0))	Edit
REPEAT_TYPE_ID		DoTS::RepeatType (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED	no	NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
NAME		STRING(255)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::PredictedAAFeature (subclass of DoTS::AAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_Feature_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
Feature_NAME_ID		DoTS::FeatureName (NUMBER(10,0))	Edit
PARENT_ID		DoTS::AAFeature (NUMBER(10,0))	Edit
NA_Feature_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
DESCRIPTION		STRING(4000)	Edit
PFAM_ENTRY_ID		DoTS::PfamEntry (NUMBER(10,0))	Edit
MOTIF_AA_SEQUENCE_ID		DoTS::AASequence (NUMBER(10,0))	Edit
REPEAT_TYPE_ID		DoTS::RepeatType (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED	no	NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
NAME		STRING(255)	Edit
ALGORITHM_NAME		STRING(255)	Edit
SCORE		FLOAT	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::RepeatRegionAAFeature (subclass of DoTS::AAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_Feature_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
Feature_NAME_ID		DoTS::FeatureName (NUMBER(10,0))	Edit
PARENT_ID		DoTS::AAFeature (NUMBER(10,0))	Edit
NA_Feature_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
DESCRIPTION		STRING(4000)	Edit
PFAM_ENTRY_ID		DoTS::PfamEntry (NUMBER(10,0))	Edit
MOTIF_AA_SEQUENCE_ID		DoTS::AASequence (NUMBER(10,0))	Edit
REPEAT_TYPE_ID		DoTS::RepeatType (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED	no	NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
PERIOD		NUMBER(12,0)	Edit
COPYNUM		FLOAT	Edit
CONSENSUS_SIZE		NUMBER(12,0)	Edit
PERCENT_MATCH		NUMBER(3,0)	Edit
PERCENT_INDEL		NUMBER(3,0)	Edit
SCORE		FLOAT	Edit
CONSENSUS		STRING(4000)	Edit
ALGORITHM_NAME		STRING(255)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> DoTS::SecondaryStructureAAFeature (subclass of [DoTS::AAFeature](#))

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_Feature_ID	no	NUMBER(10,0)	[Edit](#)
AA_SEQUENCE_ID	no	[DoTS::AASequence](#) (NUMBER(10,0))	[Edit](#)
Feature_NAME_ID		[DoTS::FeatureName](#) (NUMBER(10,0))	[Edit](#)
PARENT_ID		[DoTS::AAFeature](#) (NUMBER(10,0))	[Edit](#)
NA_Feature_ID		[DoTS::NAFeature](#) (NUMBER(10,0))	[Edit](#)
SUBCLASS_VIEW		STRING(30)	[Edit](#)
SEQUENCE_ONTOLOGY_ID		[SRes::SequenceOntology](#) (NUMBER(10,0))	[Edit](#)
DESCRIPTION		STRING(4000)	[Edit](#)
PFAM_ENTRY_ID		[DoTS::PfamEntry](#) (NUMBER(10,0))	[Edit](#)
MOTIF_AA_SEQUENCE_ID		[DoTS::AASequence](#) (NUMBER(10,0))	[Edit](#)
REPEAT_TYPE_ID		[DoTS::RepeatType](#) (NUMBER(10,0))	[Edit](#)
EXTERNAL_DATABASE_RELEASE_ID		[SRes::ExternalDatabaseRelease](#) (NUMBER(10,0))	[Edit](#)
SOURCE_ID		STRING(50)	[Edit](#)
PREDICTION_ALGORITHM_ID		[Core::Algorithm](#) (NUMBER(5,0))	[Edit](#)
IS_PREDICTED	no	NUMBER(1,0)	[Edit](#)
REVIEW_STATUS_ID		[SRes::ReviewStatus](#) (NUMBER(12,0))	[Edit](#)
NAME		STRING(255)	[Edit](#)
SCORE		FLOAT	[Edit](#)

Child tables:

Subclasses:

[webmaster@gusdb.org](#)

GUS Schema >> DoTS::SignalPeptideFeature (subclass of DoTS::AAFeature)

This view of AAFeatureImp stores predicted and annotated signal peptides.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_Feature_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
Feature_NAME_ID		DoTS::FeatureName (NUMBER(10,0))	Edit
PARENT_ID		DoTS::AAFeature (NUMBER(10,0))	Edit
NA_Feature_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
DESCRIPTION		STRING(4000)	Edit
PFAM_ENTRY_ID		DoTS::PfamEntry (NUMBER(10,0))	Edit
MOTIF_AA_SEQUENCE_ID		DoTS::AASequence (NUMBER(10,0))	Edit
REPEAT_TYPE_ID		DoTS::RepeatType (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED	no	NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
NAME		STRING(255)	Edit
ALGORITHM_NAME		STRING(255)	Edit
MAXY_SCORE		FLOAT	Edit
MAXY_CONCLUSION		NUMBER(3,0)	Edit
MAXC_SCORE		FLOAT	Edit
MAXC_CONCLUSION		NUMBER(3,0)	Edit
MAXS_SCORE		FLOAT	Edit
MAXS_CONCLUSION		NUMBER(3,0)	Edit
MEANS_SCORE		FLOAT	Edit
MEANS_CONCLUSION		NUMBER(3,0)	Edit
NUM_POSITIVES		NUMBER(12,0)	Edit
SIGNAL_PROBABILITY		FLOAT	Edit
ANCHOR_PROBABILITY		FLOAT	probability given by SignalP software Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::TertiaryStructureAAFeature (subclass of DoTS::AAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_Feature_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
Feature_NAME_ID		DoTS::FeatureName (NUMBER(10,0))	Edit
PARENT_ID		DoTS::AAFeature (NUMBER(10,0))	Edit
NA_Feature_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
DESCRIPTION		STRING(4000)	Edit
PFAM_ENTRY_ID		DoTS::PfamEntry (NUMBER(10,0))	Edit
MOTIF_AA_SEQUENCE_ID		DoTS::AASequence (NUMBER(10,0))	Edit
REPEAT_TYPE_ID		DoTS::RepeatType (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED	no	NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
NAME		STRING(255)	Edit
SCORE		FLOAT	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::TranslatedAAFeatSeg

column	nulls?	type	description
TRANSLATED_AA_FEAT_SEG_ID	no	NUMBER(10,0)	Edit
AA_Feature_ID	no	DoTS::AAFeature (NUMBER(10,0))	Edit
TRANSLATION_SCORE	no	FLOAT	Edit
START_POS	no	NUMBER(12,0)	Edit
END_POS	no	NUMBER(12,0)	Edit
AA_START_POS		NUMBER(12,0)	Edit
AA_END_POS		NUMBER(12,0)	Edit
NUCLEOTIDES_SHIFTED		NUMBER(12,0)	Edit
TYPE_OF_SHIFT		STRING(20)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::TranslatedAAFeature (subclass of DoTS::AAFeature)

the table for translated na_sequence

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_Feature_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
Feature_NAME_ID		DoTS::FeatureName (NUMBER(10,0))	Edit
PARENT_ID		DoTS::AAFeature (NUMBER(10,0))	Edit
NA_Feature_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
DESCRIPTION		STRING(4000)	Edit
PFAM_ENTRY_ID		DoTS::PfamEntry (NUMBER(10,0))	Edit
MOTIF_AA_SEQUENCE_ID		DoTS::AASequence (NUMBER(10,0))	Edit
REPEAT_TYPE_ID		DoTS::RepeatType (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED	no	NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
TRANSLATION_START		NUMBER(12,0)	Start of the translation in nucleic acid sequence Edit
TRANSLATION_STOP		NUMBER(12,0)	end of the translation in nucleic acid sequence Edit
IS_SIMPLE		NUMBER(3,0)	Edit
CODON_TABLE		STRING(255)	Edit
TRANSLATION_SCORE		FLOAT	Edit
TRANSLATION_MODEL		STRING(255)	Edit
NUMBER_OF_SEGMENTS		NUMBER(12,0)	Edit
DIANA_ATG_SCORE		FLOAT	Edit
DIANA_ATG_POSITION		NUMBER(12,0)	Edit
P_VALUE		FLOAT	Edit
Parameter_VALUES		STRING(255)	Edit
IS_REVERSED		NUMBER(3,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

[GUS Schema](#) >> DoTS::AALocation

column	nulls?	type	description
AA_LOCATION_ID	no	NUMBER(10,0)	Edit
AA_Feature_ID	no	[DoTS::AAFeature](#) (NUMBER(10,0))	Edit
START_MIN		NUMBER(12,0)	Edit
START_MAX		NUMBER(12,0)	Edit
END_MIN		NUMBER(12,0)	Edit
END_MAX		NUMBER(12,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::DbRefNAFeature

Links DbRef to NAFeature entries.

column	nulls?	type	description
DB_REF_NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_Feature_ID	no	DoTS::NAFeature (NUMBER(10,0))	Edit
DB_REF_ID	no	SRes::DbRef (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

GUS Schema >> DoTS::ExonLocation

column	nulls?	type	description
EXON_LOCATION_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
START_MIN		NUMBER(12,0)	Edit
START_MAX		NUMBER(12,0)	Edit
END_MIN		NUMBER(12,0)	Edit
END_MAX		NUMBER(12,0)	Edit
ORDER_NUM		NUMBER(5,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::FeatureName

GenBank feature names

column	nulls?	type	description
Feature_NAME_ID	no	NUMBER(10,0)	Primary key column Edit
NAME	no	STRING(30)	The name of the feature Edit
Feature_CATEGORY		STRING(30)	If the feature has a catagory place it here Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

AA Sequence Features
   DoTS::AAFeature

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::GeneFeatureSAGETagLink

column	nulls?	type	description
GENE_Feature_SAGE_TAG_LINK_ID	no	NUMBER(12,0)	Edit
GENOMIC_NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(12,0))	Edit
GENE_NA_Feature_ID	no	DoTS::NAFeature (NUMBER(12,0))	Edit
TAG_NA_Feature_ID	no	DoTS::NAFeature (NUMBER(12,0))	Edit
FIVE_PRIME_TAG_OFFSET	no	NUMBER(12,0)	Edit
THREE_PRIME_TAG_OFFSET	no	NUMBER(12,0)	Edit
SAME_STRAND	no	NUMBER(1,0)	Edit
EXPERIMENTALLY_VERIFIED	no	NUMBER(1,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::NAFeatRelationshipType

column	nulls?	type	description
NA_FEAT_RELATIONSHIP_TYPE_ID	no	NUMBER(10,0)	Edit
NAME	no	STRING(255)	Edit
DESCRIPTION	no	CLOB	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Feature Relations
   DoTS::NAFeatureRelationship

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> DoTS::NAFeatureComment

column	nulls?	type	description
NA_Feature_COMMENT_ID	no	NUMBER(10,0)	[Edit](#)
NA_Feature_ID	no	[DoTS::NAFeature](#) (NUMBER(10,0))	[Edit](#)
COMMENT_STRING	no	CLOB	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::NAFeatureRelationship

column	nulls?	type	description
NA_Feature_RELATIONSHIP_ID	no	NUMBER(10,0)	Edit
PARENT_NA_Feature_ID	no	DoTS::NAFeature (NUMBER(10,0))	Edit
CHILD_NA_Feature_ID	no	DoTS::NAFeature (NUMBER(10,0))	Edit
NA_FEAT_RELATIONSHIP_TYPE_ID	no	DoTS::NAFeatRelationshipType (NUMBER(10,0))	Edit
ORDINAL		NUMBER(10,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> DoTS::NALocation

column	nulls?	type	description
NA_LOCATION_ID	no	NUMBER(10,0)	Edit
NA_Feature_ID	no	[DoTS::NAFeature](#) (NUMBER(10,0))	Edit
START_MIN		NUMBER(12,0)	Edit
START_MAX		NUMBER(12,0)	Edit
END_MIN		NUMBER(12,0)	Edit
END_MAX		NUMBER(12,0)	Edit
LOC_ORDER		NUMBER(12,0)	Edit
IS_REVERSED		NUMBER(3,0)	Edit
IS_EXCLUDED		NUMBER(3,0)	Edit
DB_NAME		STRING(100)	Edit
DB_IDENTIFIER		STRING(25)	Edit
LITERAL_SEQUENCE		STRING(255)	Edit
LOCATION_TYPE		STRING(50)	Edit
REMARK		STRING(255)	Edit
DEBUG_FIELD		STRING(4000)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> **DoTS::RNAFeatureExon**

column	nulls?	type	description
RNA_Feature_EXON_ID	no	NUMBER(10,0)	[Edit](#)
RNA_Feature_ID	no	[DoTS::NAFeature](#) (NUMBER(10,0))	[Edit](#)
EXON_Feature_ID	no	[DoTS::NAFeature](#) (NUMBER(10,0))	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::AAComment

Comments about protein

column	nulls?	type	description
AA_COMMENT_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Foreign key to AASequenceImp table Edit
COMMENT_NAME_ID		DoTS::CommentName (NUMBER(4,0))	Foreign key to CommentName Edit
COMMENT_STRING		STRING(4000)	Description of amino acid sequence Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::AAEntry

column	nulls?	type	description
AA_ENTRY_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
SOURCE_ID	no	STRING(15)	Edit
STATUS		STRING(12)	Edit
CREATED_DATE		DATE	Edit
CREATED_REL_VER		NUMBER(5,0)	Edit
SEQ_DATE		DATE	Edit
SEQ_REL_VER		NUMBER(5,0)	Edit
ANNOT_DATE		DATE	Edit
ANNOT_REL_VER		NUMBER(5,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

GenBank Sequence Record
   DoTS::SecondaryAccs

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::AAGene

column	nulls?	type	description
AA_GENE_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
NAME	no	STRING(50)	Edit
IS_VERIFIED	no	NUMBER(1,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

GenBank Sequence Record
   DoTS::AAGeneSynonym

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::AAGeneSynonym

column	nulls?	type	description
AA_GENE_SYNONYM_ID	no	NUMBER(10,0)	Edit
AA_GENE_ID	no	DoTS::AAGene (NUMBER(10,0))	Edit
SYNONYM_NAME	no	STRING(50)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::AASequenceKeyword

column	nulls?	type	description
AA_SEQUENCE_KEYWORD_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
KEYWORD_ID	no	DoTS::Keyword (NUMBER(6,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

GUS Schema >> DoTS::AASequenceOrganelle

column	nulls?	type	description
AA_SEQUENCE_ORGANELLE_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
ORGANELLE_ID	no	DoTS::Organelle (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::AASequenceRef

column	nulls?	type	description
AA_SEQUENCE_REF_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
REFERENCE_ID	no	SRes::Reference (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::AASequenceTaxon

column	nulls?	type	description
AA_SEQUENCE_TAXON_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
TAXON_ID	no	SRes::Taxon (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::CommentName

column	nulls?	type	description
COMMENT_NAME_ID	no	NUMBER(4,0)	Edit
NAME	no	STRING(20)	Edit
C_CATEGORY		STRING(20)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

GenBank Sequence Record
   DoTS::AAComment

Subclasses:

---

webmaster@gusdb.org

[GUS Schema](#) >> DoTS::Keyword

column	nulls?	type	description
KEYWORD_ID	no	NUMBER(6,0)	[Edit](#)
KEYWORD	no	STRING(80)	[Edit](#)
DESCRIPTION		STRING(255)	[Edit](#)
PREFERRED_KEYWORD_ID		[DoTS::Keyword](#) (NUMBER(6,0))	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

GenBank Sequence Record
    [DoTS::AASequenceKeyword](#)
    [DoTS::Keyword](#)
    [DoTS::NASequenceKeyword](#)
Uncategorized
    [DoTS::ExternalDatabaseKeyword](#)

Subclasses:

---

webmaster@gusdb.org

[GUS Schema](#) >> **DoTS::NAComment**

column	nulls?	type	description
NA_COMMENT_ID	no	NUMBER(10,0)	[Edit](#)
NA_SEQUENCE_ID	no	[DoTS::NASequence](#) (NUMBER(10,0))	[Edit](#)
COMMENT_STRING	no	CLOB	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

Subclasses:

GUS Schema >> DoTS::NAEntry

column	nulls?	type	description
NA_ENTRY_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(10,0))	Edit
SOURCE_ID	no	STRING(15)	Edit
STATUS		STRING(12)	Edit
DIVISION		STRING(15)	Edit
VERSION		NUMBER(12,0)	Edit
NID		STRING(15)	Edit
CREATED_DATE		DATE	Edit
CREATED_REL_VER		NUMBER(5,0)	Edit
UPDATE_DATE		DATE	Edit
UPDATE_REL_VER		NUMBER(5,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

GenBank Sequence Record
   DoTS::SecondaryAccs

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> **DoTS::NAFeatureNAGene**

column	nulls?	type	description
NA_Feature_NA_GENE_ID	no	NUMBER(10,0)	Edit
NA_GENE_ID	no	[DoTS::NAGene](#) (NUMBER(10,0))	Edit
NA_Feature_ID	no	[DoTS::NAFeature](#) (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::NAFeatureNAProtein

column	nulls?	type	description
NA_Feature_NA_PROTEIN_ID	no	NUMBER(10,0)	Edit
NA_PROTEIN_ID	no	DoTS::NAProtein (NUMBER(10,0))	Edit
NA_Feature_ID	no	DoTS::NAFeature (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

GUS Schema >> DoTS::NAFeatureNAPT

column	nulls?	type	description
NA_Feature_NA_PT_ID	no	NUMBER(10,0)	Edit
NA_PRIMARY_TRANSCRIPT_ID	no	DoTS::NAPrimaryTranscript (NUMBER(10,0))	Edit
NA_Feature_ID	no	DoTS::NAFeature (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::NAGene

column	nulls?	type	description
NA_GENE_ID	no	NUMBER(10,0)	Edit
NAME		STRING(300)	Edit
IS_VERIFIED	no	NUMBER(1,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

GenBank Sequence Record
   DoTS::NAFeatureNAGene
   DoTS::NAPrimaryTranscript

Subclasses:

---

gusdb.org | Schema Browser

GUS Schema >> DoTS::NAPrimaryTranscript

column	nulls?	type	description
NA_PRIMARY_TRANSCRIPT_ID	no	NUMBER(10,0)	Edit
NA_GENE_ID		DoTS::NAGene (NUMBER(10,0))	Edit
IS_VERIFIED	no	NUMBER(1,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

GenBank Sequence Record
   DoTS::NAFeatureNAPT
   DoTS::NAProtein

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::NAProtein

column	nulls?	type	description
NA_PROTEIN_ID	no	NUMBER(10,0)	Edit
NA_PRIMARY_TRANSCRIPT_ID		DoTS::NAPrimaryTranscript (NUMBER(10,0))	Edit
NAME		STRING(300)	Edit
IS_VERIFIED	no	NUMBER(1,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

GenBank Sequence Record
    DoTS::NAFeatureNAProtein

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::NASeqCytoLocation

column	nulls?	type	description
NA_SEQ_CYTO_LOCATION_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(10,0))	Edit
CHROMOSOME	no	CHARACTER(2)	Edit
START_BAND	no	STRING(8)	Edit
END_BAND	no	STRING(8)	Edit
METHOD		STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::NASequenceKeyword

column	nulls?	type	description
NA_SEQUENCE_KEYWORD_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(10,0))	Edit
KEYWORD_ID	no	DoTS::Keyword (NUMBER(6,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::NASequenceOrganelle

column	nulls?	type	description
NA_SEQUENCE_ORGANELLE_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(10,0))	Edit
ORGANELLE_ID	no	DoTS::Organelle (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::NASequenceRef

column	nulls?	type	description
NA_SEQUENCE_REF_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(10,0))	Edit
REFERENCE_ID	no	SRes::Reference (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

[GUS Schema](#) >> DoTS::Organelle

column	nulls?	type	description
ORGANELLE_ID	no	NUMBER(10,0)	Edit
NAME		STRING(300)	Edit
PLASMID_NAME		STRING(300)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

GenBank Sequence Record
   [DoTS::AASequenceOrganelle](#)
   [DoTS::NASequenceOrganelle](#)

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::SecondaryAccs

column	nulls?	type	description
SECONDARY_ACCS_ID	no	NUMBER(10,0)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
AA_ENTRY_ID		DoTS::AAEntry (NUMBER(10,0))	Edit
NA_ENTRY_ID		DoTS::NAEntry (NUMBER(10,0))	Edit
SOURCE_ID	no	STRING(15)	Edit
SECONDARY_ACCS	no	STRING(15)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

[GUS Schema](#) >> DoTS::BLATAlignment

column	nulls?	type	description
BLAT_ALIGNMENT_ID	no	NUMBER(10,0)	Edit
QUERY_NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(10,0))	Edit
TARGET_NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(10,0))	Edit
QUERY_TABLE_ID		Core::TableInfo (NUMBER(5,0))	Edit
QUERY_TAXON_ID		SRes::Taxon (NUMBER(10,0))	Edit
QUERY_EXTERNAL_DB_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
TARGET_TABLE_ID		Core::TableInfo (NUMBER(5,0))	Edit
TARGET_TAXON_ID		SRes::Taxon (NUMBER(10,0))	Edit
TARGET_EXTERNAL_DB_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
IS_CONSISTENT	no	NUMBER(1,0)	Edit
IS_GENOMIC_CONTAMINANT	no	NUMBER(1,0)	Edit
UNALIGNED_3P_BASES	no	NUMBER(12,0)	Edit
UNALIGNED_5P_BASES	no	NUMBER(12,0)	Edit
HAS_3P_POLYA	no	NUMBER(1,0)	Edit
HAS_5P_POLYA	no	NUMBER(1,0)	Edit
IS_3P_COMPLETE	no	NUMBER(1,0)	Edit
IS_5P_COMPLETE	no	NUMBER(1,0)	Edit
PERCENT_IDENTITY	no	NUMBER(3,0)	Edit
MAX_QUERY_GAP	no	NUMBER(12,0)	Edit
MAX_TARGET_GAP	no	NUMBER(12,0)	Edit
NUMBER_OF_SPANS	no	NUMBER(5,0)	Edit
QUERY_START	no	NUMBER(12,0)	Edit
QUERY_END	no	NUMBER(12,0)	Edit
TARGET_START	no	NUMBER(12,0)	Edit
TARGET_END	no	NUMBER(12,0)	Edit
IS_REVERSED	no	NUMBER(1,0)	Edit
QUERY_BASES_ALIGNED	no	NUMBER(12,0)	Edit
REPEAT_BASES_ALIGNED	no	NUMBER(12,0)	Edit
NUM_NS	no	NUMBER(12,0)	Edit
SCORE	no	FLOAT	Edit
IS_BEST_ALIGNMENT	no	NUMBER(1,0)	Edit
BLAT_ALIGNMENT_QUALITY_ID	no	DoTS::BLATAlignmentQuality (NUMBER(12,0))	Edit
BLOCKSIZES	no	STRING(4000)	Edit
QSTARTS	no	STRING(4000)	Edit
TSTARTS	no	STRING(4000)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Misc Applications
  DoTS::GeneFeatureSeqOverlap

Subclasses:

GUS Schema >> DoTS::BLATAlignmentQuality

column	nulls?	type	description
BLAT_ALIGNMENT_QUALITY_ID	no	NUMBER(12,0)	Edit
NAME	no	STRING(30)	Edit
DESCRIPTION	no	STRING(4000)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Similarity
    DoTS::BLATAlignment

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::ConsistentAlignment

column	nulls?	type	description
CONSISTENT_ALIGNMENT_ID	no	NUMBER(10,0)	Edit
SIMILARITY_ID	no	DoTS::Similarity (NUMBER(10,0))	Edit
GENOMIC_NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(10,0))	Edit
TRANSCRIPT_NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(10,0))	Edit
IS_3P_COMPLETE	no	NUMBER(1,0)	Edit
IS_5P_COMPLETE	no	NUMBER(1,0)	Edit
IS_CONSISTENT	no	NUMBER(1,0)	Edit
IS_GENOMIC_CONTAMINANT	no	NUMBER(1,0)	Edit
MIN_SPAN_IDENTITY	no	NUMBER(3,0)	Edit
AVG_IDENTITY	no	NUMBER(3,0)	Edit
MAX_QUERY_GAP	no	NUMBER(12,0)	Edit
DOTS_BASES_ALIGNED	no	NUMBER(12,0)	Edit
NUMBER_OF_SPANS	no	NUMBER(5,0)	Edit
DOTS_START	no	NUMBER(12,0)	Edit
DOTS_END	no	NUMBER(12,0)	Edit
GENOMIC_START	no	NUMBER(12,0)	Edit
GENOMIC_END	no	NUMBER(12,0)	Edit
IS_REVERSED	no	NUMBER(1,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::Similarity

"Table to record Similarities between two identifers in the database (BLAST, FASTA, SIM4 etc)"

column	nulls?	type	description
SIMILARITY_ID	no	NUMBER(10,0)	primary_key Edit
SUBJECT_TABLE_ID	no	Core::TableInfo (NUMBER(5,0))	"table_id of the subject part of the similarity (ie, the entry from the database)" Edit
SUBJECT_ID	no	NUMBER(10,0)	row_id of the subject Edit
QUERY_TABLE_ID	no	Core::TableInfo (NUMBER(5,0))	table_id of the query part of the similarity Edit
QUERY_ID	no	NUMBER(10,0)	row_id of the query Edit
SCORE	no	FLOAT	score of the similarity Edit
BIT_SCORE_SUMMARY		FLOAT	Edit
PVALUE_MANT	no	FLOAT	P value mantissa Edit
PVALUE_EXP	no	NUMBER(6,0)	P value exponent Edit
MIN_SUBJECT_START	no	NUMBER(12,0)	minimum location in the subject of the match Edit
MAX_SUBJECT_END	no	NUMBER(12,0)	maximum location in the subject of the match Edit
MIN_QUERY_START	no	NUMBER(12,0)	minimum location in the query of the match Edit
MAX_QUERY_END	no	NUMBER(12,0)	maximum location in the query of the match Edit
NUMBER_OF_MATCHES	no	NUMBER(5,0)	number of spans (HSPs for BLAST) that are involved in this similarity Edit
TOTAL_MATCH_LENGTH	no	NUMBER(12,0)	total length of all the matches Edit
NUMBER_IDENTICAL	no	NUMBER(12,0)	total number of identical matches Edit
NUMBER_POSITIVE	no	NUMBER(12,0)	total number of positives (conservative replacements) Edit
IS_REVERSED	no	NUMBER(1,0)	matches on the reverse strand Edit
READING_FRAME		NUMBER(3,0)	reading frame of this Similarity if relevant (BLASTX etc) Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Similarity
    DoTS::ConsistentAlignment
    DoTS::SimilaritySpan
Misc Applications
    DoTS::GeneFeatureSeqOverlap

Subclasses:

gusdb.org | Schema Browser

GUS Schema >> DoTS::SimilaritySpan

Table to record the specifics for each span (HSP) in a similarity

column	nulls?	type	description
SIMILARITY_SPAN_ID	no	NUMBER(10,0)	primary_key Edit
SIMILARITY_ID	no	DoTS::Similarity (NUMBER(10,0))	foreign key to Similarity Edit
MATCH_LENGTH	no	NUMBER(12,0)	length of the match Edit
NUMBER_IDENTICAL	no	NUMBER(12,0)	number of identical matches Edit
NUMBER_POSITIVE	no	NUMBER(12,0)	number of postives allowing for conservative replacements Edit
SCORE	no	FLOAT	score of this span Edit
BIT_SCORE		FLOAT	Edit
PVALUE_MANT	no	FLOAT	P value mantissa Edit
PVALUE_EXP	no	NUMBER(6,0)	P value exponent Edit
SUBJECT_START		NUMBER(12,0)	position in subject where this match starts Edit
SUBJECT_END		NUMBER(12,0)	position in subject where this match ends Edit
QUERY_START		NUMBER(12,0)	position in query where this match starts Edit
QUERY_END		NUMBER(12,0)	position in query there this match ends Edit
IS_REVERSED	no	NUMBER(1,0)	if true then this match is on the reverse strand Edit
READING_FRAME		NUMBER(3,0)	reading frame of the match if relevant (BLASTX etc) Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

http://www.gusdb.org/SchemaBrowser/table.htm?schema=DoTS&table=SimilaritySpan[2/26/13 3:53:47 PM]

GUS Schema >> DoTS::AssemblyAnatomyPercent

Materialized view that maps Assemblies to the Anatomy ontology based on source library of the contained sequences

column	nulls?	type	description
ASSEMBLY_ANATOMY_PERCENT_ID	no	NUMBER(10,0)	primary key Edit
NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(10,0))	FK to the Assembly Edit
TAXON_ID		SRes::Taxon (NUMBER(12,0))	Edit
ANATOMY_ID	no	SRes::Anatomy (NUMBER(4,0))	FK to the Anatomy table Edit
PERCENT	no	FLOAT	Percent of informative AssemblySequences that come from this anatomy Edit
ANATOMY_ESTS	no	NUMBER(5,0)	Edit
EST_COUNT	no	NUMBER(5,0)	number of informative sequences (have a library anatomy assignment) in this assembly Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

gusdb.org | Schema Browser

GUS Schema >> DoTS::AssemblySequence

Table to hold the specific information about sequences contained in Assemblies

column	nulls?	type	description
ASSEMBLY_SEQUENCE_ID	no	NUMBER(10,0)	Primary key Edit
ASSEMBLY_NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	foreign key to the Assembly table Edit
NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(10,0))	foreign key to the ExternalNASequence table...identifies the input sequence Edit
SEQUENCE_START	no	NUMBER(12,0)	position in the ExternalNASequence where the AssemblySequence starts Edit
SEQUENCE_END	no	NUMBER(12,0)	position in the ExternalNASequence where the AssemblySequence ends Edit
QUALITY_START		NUMBER(12,0)	position in the ExternalNASequence where the cap4 determines the quality sequence starts Edit
QUALITY_END		NUMBER(12,0)	position in the ExternalNASequence where the cap4 determines the quality sequence ends Edit
ASSEMBLY_OFFSET	no	NUMBER(12,0)	position in the assembly where this AssemblySequence begins Edit
ASSEMBLY_STRAND	no	NUMBER(1,0)	Orientation of this AssemblySequence in the Assembly (0|1) Edit
GAPPED_SEQUENCE		CLOB	Edit
HAVE_PROCESSED	no	NUMBER(1,0)	This sequence has been processed for assembly Edit
PROCESSED_CATEGORY		STRING(50)	Category of this processed sequence ('repeat','chimera','low_quality') if not included in an Assembly Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Assembly
   DoTS::AssemblySequenceSNP

Subclasses:

webmaster@gusdb.org

http://www.gusdb.org/SchemaBrowser/table.htm?schema=DoTS&table=AssemblySequence[2/26/13 3:54:00 PM]

GUS Schema >> DoTS::AssemblySequenceSNP

table that records the specifics of the putative SNP in this AssemblySequence

column	nulls?	type	description
ASSEMBLY_SEQUENCE_SNP_ID	no	NUMBER(10,0)	primary_key Edit
ASSEMBLY_SEQUENCE_ID	no	DoTS::AssemblySequence (NUMBER(10,0))	foreign key to AssemblySequence Edit
ASSEMBLY_SNP_ID	no	DoTS::AssemblySNP (NUMBER(10,0))	foreign key to AssemblySNP Edit
GAPPED_LOCATION	no	NUMBER(8,0)	position in the gapped AssemblySequence of this SNP Edit
NA_SEQUENCE_LOCATION		NUMBER(8,0)	position in the NASequence from which this AssemblySequence is derived of this SNP Edit
SEQUENCE_CHARACTER	no	CHARACTER(1)	the character at this position in this sequence. Intended to facilitate haplotype identification. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::AssemblySNP

Table to record information about SNPs found in Assemblies

column	nulls?	type	description
ASSEMBLY_SNP_ID	no	NUMBER(10,0)	primary_key Edit
NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(10,0))	relation to the Assembly table Edit
NUM_A	no	NUMBER(8,0)	number of A bases in this column Edit
NUM_C	no	NUMBER(8,0)	number of C bases in this column Edit
NUM_G	no	NUMBER(8,0)	number of G bases in this column Edit
NUM_T	no	NUMBER(8,0)	number of T bases in this column Edit
NUM_OTHER	no	NUMBER(8,0)	numbe of other bases (such as N) in this column Edit
NUC_1	no	CHARACTER(1)	predominant nucleotide (ACGTN) Edit
NUC_2		CHARACTER(1)	second most prevalent nucleotide Edit
NUC_3		CHARACTER(1)	third most prevalent nucleotide Edit
NUC_4		CHARACTER(1)	fourth most prevalent nucleotide Edit
NUM_NUC_1	no	NUMBER(8,0)	number of most prevalent nucleotide Edit
NUM_NUC_2		NUMBER(8,0)	number of second most prevalent nucleotide Edit
NUM_NUC_3		NUMBER(8,0)	number of third most prevalent nucleotide Edit
NUM_NUC_4		NUMBER(8,0)	number of fourth most prevalent nucleotide Edit
FRAction_NUC_1	no	FLOAT	fraction that represents the most prevalent nucleotide (number most prevalent / Assembly depth) Edit
NUM_DISTINCT_ACGT	no	NUMBER(1,0)	number of distinct different nuclotides at this position Edit
ASSEMBLY_DEPTH	no	NUMBER(5,0)	depth (number of sequences at this position Edit
ASSEMBLY_POSITION	no	NUMBER(8,0)	position in the assembly (including gaps) of this SNP Edit
CONSENSUS_POSITION	no	NUMBER(8,0)	position in the consensus sequence of this SNP Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Assembly
   DoTS::AssemblySequenceSNP

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::GeneTrapAssembly

column	nulls?	type	description
GENE_TRAP_ASSEMBLY_ID	no	NUMBER(10,0)	Edit
TAG_NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(10,0))	Edit
ASSEMBLY_NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(10,0))	Edit
IS_BEST_MATCH	no	NUMBER(1,0)	Edit
MATCH_START	no	NUMBER(10,0)	Edit
MATCH_END	no	NUMBER(10,0)	Edit
IS_REVERSED	no	NUMBER(1,0)	Edit
PERCENT_IDENTITY	no	FLOAT	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::MergeSplit

Table to record when entries (primarily Assemblies or RNAs) merge or split due to Assembly updates

column	nulls?	type	description
MERGE_SPLIT_ID	no	NUMBER(10,0)	primary_key Edit
TABLE_ID	no	Core::TableInfo (NUMBER(5,0))	table_id of entries that have merged/split Edit
OLD_ID	no	NUMBER(10,0)	old identifier in this table Edit
NEW_ID	no	NUMBER(10,0)	new identifier in this table Edit
MERGE_SPLIT_GROUP_ID	no	NUMBER(12,0)	"if multiple entries are merged in a single event, this serves to group them together" Edit
IS_MERGE	no	NUMBER(1,0)	"if true then is a merge, Not a split" Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::Motif

column	nulls?	type	description
MOTIF_ID	no	NUMBER(10,0)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID	no	STRING(15)	Edit
NAME	no	STRING(80)	Edit
MOTIF	no	STRING(255)	Edit
DESCRIPTION	no	STRING(255)	Edit
PROBABILITY	no	FLOAT	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::MotifRejectionReason

column	nulls?	type	description
MOTIF_REJECTION_REASON_ID	no	NUMBER(10,0)	Edit
NAME	no	STRING(255)	Edit
DESCRIPTION		STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Motifs
    DoTS::RejectedMotif

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::RejectedMotif

column	nulls?	type	description
REJECTED_MOTIF_ID	no	NUMBER(10,0)	Edit
SOURCE_ID	no	STRING(32)	Edit
EXTERNAL_DATABASE_ID	no	SRes::ExternalDatabase (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID	no	SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
MOTIF_REJECTION_REASON_ID		DoTS::MotifRejectionReason (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::Gene

A central dogma (GENE->RNA->PROTEIN) table containing gene information for the genes in GUS

column	nulls?	type	description
GENE_ID	no	NUMBER(10,0)	primary key Edit
NAME		STRING(255)	The HUGO approved full gene name for human genes or the MGI approved full gene name for mouse genes Edit
GENE_SYMBOL		STRING(50)	"The HUGO or MGI approved abbreviated gene name or symbol (e.g., Kit)" Edit
GENE_CATEGORY_ID		DoTS::GeneCategory (NUMBER(10,0))	The identifier of a gene category that has been assigned to the gene Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	The identifier of the review status that has been assigned to the gene Edit
DESCRIPTION		STRING(500)	A description of the gene Edit
REVIEWER_SUMMARY		STRING(4000)	Annotator comments regarding the gene Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Central Dogma
DoTS::GeneChromosomalLocation
DoTS::GeneInstance
DoTS::GeneSynonym
DoTS::RNA
Variation
DoTS::Allele
Uncategorized
DoTS::FamilyGene
DoTS::GeneFamilyRelation

Subclasses:

GUS Schema >> DoTS::GeneCategory

A controlled vocabulary of the gene categories that could be assigned to a gene

column	nulls?	type	description
GENE_CATEGORY_ID	no	NUMBER(10,0)	primary key Edit
TERM	no	STRING(80)	"A category (e.g., pseudogene) that could be assigned to a gene" Edit
DEFINITION		STRING(255)	A definition of the category Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Central Dogma
   DoTS::Gene

Subclasses:

---

webmaster@gusdb.org

[GUS Schema](#) >> **DoTS::GeneChromosomalLocation**

column	nulls?	type	description
GENE_CHROMOSOMAL_LOCATION_ID	no	NUMBER(10,0)	Edit
GENE_ID	no	DoTS::Gene (NUMBER(10,0))	Edit
CHROMOSOME	no	NUMBER(5,0)	Edit
CYTOGENETIC		STRING(50)	Edit
METHOD		STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::GeneInstance

column	nulls?	type	description
GENE_INSTANCE_ID	no	NUMBER(10,0)	Edit
GENE_INSTANCE_CATEGORY_ID		DoTS::GeneInstanceCategory (NUMBER(3,0))	Edit
GENE_ID	no	DoTS::Gene (NUMBER(10,0))	Edit
NA_Feature_ID	no	DoTS::NAFeature (NUMBER(10,0))	Edit
DESCRIPTION		STRING(255)	Edit
REVIEWER_SUMMARY		STRING(4000)	Edit
IS_REFERENCE	no	NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::GeneInstanceCategory

column	nulls?	type	description
GENE_INSTANCE_CATEGORY_ID	no	NUMBER(3,0)	Edit
NAME	no	STRING(40)	Edit
DESCRIPTION		STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Central Dogma
  DoTS::GeneInstance

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::GeneSynonym

This table stores gene synonym.

column	nulls?	type	description
GENE_SYNONYM_ID	no	NUMBER(10,0)	primary key Edit
GENE_ID	no	DoTS::Gene (NUMBER(10,0))	The identifier of the gene that has been assigned the gene symbol Edit
SYNONYM_NAME	no	STRING(30)	The gene symbol Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	The identifer of the review status Edit
IS_OBSOLETE		NUMBER(1,0)	Indicates if this synonym is no longer used Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> **DoTS::Protein**

A central dogma (GENE->RNA->PROTEIN) table containing Protein information for GUS proteins

column	nulls?	type	description
PROTEIN_ID	no	NUMBER(18,0)	primary key Edit
NAME		STRING(255)	"The name of a protein (e.g., P450 monooxygenase)" Edit
RNA_ID	no	DoTS::RNA (NUMBER(10,0))	The identifier of the rna which results in the protein entry Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	The identifier of the review status Edit
DESCRIPTION		STRING(500)	A description of the protein Edit
REVIEWER_SUMMARY		STRING(4000)	Annotator comments regarding the Protein Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Central Dogma
    DoTS::ProteinInstance
    DoTS::ProteinProteinCategory
    DoTS::ProteinSynonym
Uncategorized
    DoTS::FamilyProtein

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::ProteinCategory

A controlled vocabulary of the protein categories that could be assigned to a protein

column	nulls?	type	description
PROTEIN_CATEGORY_ID	no	NUMBER(10,0)	primary key Edit
TERM	no	STRING(80)	"A category (e.g., phosphorylated) that could be assigned to a protein" Edit
DEFINITION		STRING(255)	A definition of the category Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Central Dogma
    DoTS::ProteinProteinCategory

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::ProteinInstance

column	nulls?	type	description
PROTEIN_INSTANCE_ID	no	NUMBER(10,0)	Edit
PROTEIN_INSTANCE_CATEGORY_ID		DoTS::ProteinInstanceCategory (NUMBER(3,0))	Edit
PROTEIN_ID	no	DoTS::Protein (NUMBER(18,0))	Edit
AA_Feature_ID	no	DoTS::AAFeature (NUMBER(10,0))	Edit
REVIEWER_SUMMARY		STRING(4000)	Edit
IS_REFERENCE	no	NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::ProteinInstanceCategory

column	nulls?	type	description
PROTEIN_INSTANCE_CATEGORY_ID	no	NUMBER(3,0)	Edit
NAME	no	STRING(40)	Edit
DESCRIPTION		STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Central Dogma
   DoTS::ProteinInstance

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::ProteinProperty

column	nulls?	type	description
PROTEIN_PROPERTY_ID	no	NUMBER(10,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(10,0))	Edit
PROTEIN_PROPERTY_TYPE_ID	no	DoTS::ProteinPropertyType (NUMBER(10,0))	Edit
VALUE	no	FLOAT	Edit
VALUE_UNITS		STRING(15)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
SOURCE_ID		STRING(32)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

gusdb.org | Schema Browser

GUS Schema >> DoTS::ProteinPropertyType

column	nulls?	type	description
PROTEIN_PROPERTY_TYPE_ID	no	NUMBER(10,0)	Edit
NAME	no	STRING(80)	Edit
DESCRIPTION	no	STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Central Dogma
   DoTS::ProteinProperty

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::ProteinProteinCategory

Linking table for storing Protein_ids and their Protein_category_id assignments

column	nulls?	type	description
PROTEIN_PROTEIN_CATEGORY_ID	no	NUMBER(10,0)	primary key Edit
PROTEIN_ID	no	DoTS::Protein (NUMBER(18,0))	The identifier for the protein Edit
PROTEIN_CATEGORY_ID	no	DoTS::ProteinCategory (NUMBER(10,0))	The identifier for the protein category assignment Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::ProteinSynonym

A table which contains protein names that have been assigned to a protein if multiple protein names exist for the protein entry

column	nulls?	type	description
PROTEIN_SYNONYM_ID	no	NUMBER(18,0)	primary key Edit
PROTEIN_ID	no	DoTS::Protein (NUMBER(18,0))	The identifier of the protein that has been assigned the protein name Edit
SYNONYM_NAME	no	STRING(50)	The protein name Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	The identifier of the review status Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::RNA

A central dogma (GENE->RNA->PROTEIN) table containing RNA information for the RNAs in GUS

column	nulls?	type	description
RNA_ID	no	NUMBER(10,0)	primary key Edit
DESCRIPTION		STRING(500)	The description of the RNA Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	The identifier of the review status that has been assigned to the RNA Edit
GENE_ID		DoTS::Gene (NUMBER(10,0))	The identifier of the gene to which the RNA has been assigned Edit
REVIEWER_SUMMARY		STRING(4000)	Annotator comments regarding the RNA Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Central Dogma
    DoTS::Protein
    DoTS::RNAAnatomy
    DoTS::RNAIConstruct
    DoTS::RNAInstance
    DoTS::RNARNACategory

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::RNAAnatomy

A table to store the assignments of an anatomy term to a RNA

column	nulls?	type	description
RNA_ANATOMY_ID	no	NUMBER(10,0)	primary key Edit
ANATOMY_ID	no	SRes::Anatomy (NUMBER(4,0))	The identifier of an anatomy term that has been assigned to the RNA Edit
RNA_ID	no	DoTS::RNA (NUMBER(10,0))	The identifier of the RNA Edit
EST_NUMBER		NUMBER(6,0)	The number of ESTs that have been assigned that anatomy term and are represented in the RNA Edit
RAD_CONFIRMED	no	NUMBER(1,0)	True (1) if an experiment in RAD (i.e., microarray) can confirm the expression of the RNA in the assigned anatomy Edit
LITERATURE_CONFIRMED	no	NUMBER(1,0)	True (1) if an experiment described in the literature can confirm the expression of the RNA in the assigned anatomy Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Central Dogma
   DoTS::RNAAnatomyLOE

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::RNAAnatomyLOE

Table to associate the lines of evidence with the RNAAnatomy assignments

column	nulls?	type	description
RNA_ANATOMY_LOE_ID	no	NUMBER(10,0)	primary key Edit
RNA_ANATOMY_ID	no	DoTS::RNAAnatomy (NUMBER(10,0))	foreign key to RNAAnatomy Edit
ANATOMY_LOE_ID	no	DoTS::AnatomyLOE (NUMBER(5,0))	foreign key to AnatomyLOE Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

GUS Schema >> DoTS::RNACategory

A controlled vocabulary of the RNA cateogories that could be assigned to a RNA

column	nulls?	type	description
RNA_CATEGORY_ID	no	NUMBER(10,0)	primary key Edit
TERM	no	STRING(80)	"A category (e.g., exon skipped) that could be assigned to a RNA" Edit
DEFINITION		STRING(255)	A definition of the category Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Central Dogma
DoTS::RNARNACategory

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::RNAIConstruct

column	nulls?	type	description
RNAI_CONSTRUCT_ID	no	NUMBER(10,0)	Edit
RNA_ID	no	DoTS::RNA (NUMBER(10,0))	Edit
DESCRIPTION		STRING(1000)	Edit
SEQUENCE		CLOB	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Central Dogma
DoTS::RNAIPhenotype

Subclasses:

webmaster@gusdb.org

gusdb.org | Schema Browser

GUS Schema >> DoTS::RNAInstance

column	nulls?	type	description
RNA_INSTANCE_ID	no	NUMBER(10,0)	Edit
RNA_ID	no	DoTS::RNA (NUMBER(10,0))	Edit
NA_Feature_ID	no	DoTS::NAFeature (NUMBER(10,0))	Edit
REVIEWER_SUMMARY		STRING(4000)	Edit
IS_REFERENCE	no	NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
RNA_INSTANCE_CATEGORY_ID		DoTS::RNAInstanceCategory (NUMBER(3,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::RNAInstanceCategory

column	nulls?	type	description
RNA_INSTANCE_CATEGORY_ID	no	NUMBER(3,0)	Edit
NAME	no	STRING(255)	Edit
DESCRIPTION		STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Central Dogma
DoTS::RNAInstance

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::RNAIPhenotype

column	nulls?	type	description
RNAI_PHENOTYPE_ID	no	NUMBER(10,0)	Edit
RNAI_CONSTRUCT_ID	no	DoTS::RNAIConstruct (NUMBER(10,0))	Edit
PHENOTYPE_ID	no	SRes::Phenotype (NUMBER(10,0))	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::RNARNACategory

column	nulls?	type	description
RNA_RNA_CATEGORY_ID	no	NUMBER(10,0)	primary key Edit
RNA_ID	no	DoTS::RNA (NUMBER(10,0))	The identifier for the RNA Edit
RNA_CATEGORY_ID	no	DoTS::RNACategory (NUMBER(10,0))	The identifier of the RNA category assignment Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

gusdb.org | Schema Browser

GUS Schema >> DoTS::RNASecondaryStructure (subclass of DoTS::NAFeature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
NA_Feature_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
SUBCLASS_VIEW		STRING(30)	Edit
NAME	no	STRING(30)	Edit
SEQUENCE_ONTOLOGY_ID		SRes::SequenceOntology (NUMBER(10,0))	Edit
PARENT_ID		DoTS::NAFeature (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
PREDICTION_ALGORITHM_ID		Core::Algorithm (NUMBER(5,0))	Edit
IS_PREDICTED		NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
SCORE		FLOAT	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::AAFamilyExperiment

Experiments to compute protein sequence families

column	nulls?	type	description
AA_FAMILY_EXPERIMENT_ID	no	NUMBER(12,0)	Edit
DESCRIPTION		STRING(2000)	Description about the experiment to compute protein sequene family Edit
AA_ORTHOLOG_EXPERIMENT_ID		DoTS::AASeqGroupExperiment (NUMBER(12,0))	Edit
AA_PARALOG_EXPERIMENT_ID		DoTS::AASeqGroupExperiment (NUMBER(12,0))	Foreign key to AAParalogExperiment,where the experiment of computing paralogs groups are stored Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Paralog and Family
DoTS::AASequenceFamily

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::AAParalogExperiment (subclass of DoTS::AASeqGroupExperiment)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_SEQ_GROUP_EXPERIMENT_ID	no	NUMBER(12,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
DESCRIPTION		STRING(2000)	Edit
SEQUENCE_SOURCE		STRING(500)	Edit
PVALUE_MANT		FLOAT	Edit
PVALUE_EXP		NUMBER(12,0)	Edit
PERCENT_IDENTITY		FLOAT	Edit
PERCENT_MATCH		FLOAT	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> DoTS::AASeqGroupExperiment

column	nulls?	type	description
AA_SEQ_GROUP_EXPERIMENT_ID	no	NUMBER(12,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
DESCRIPTION		STRING(2000)	Edit
SEQUENCE_SOURCE		STRING(500)	Edit
PVALUE_MANT		FLOAT	Edit
PVALUE_EXP		NUMBER(12,0)	Edit
PERCENT_IDENTITY		FLOAT	Edit
PERCENT_MATCH		FLOAT	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Paralog and Family
[DoTS::AAFamilyExperiment](#)
AA Ortholog
[DoTS::AASequenceGroup](#)

Subclasses:

[DoTS::AAParalogExperiment](#)
[DoTS::AAOrthologExperiment](#)

---

webmaster@gusdb.org

GUS Schema >> DoTS::AAParalogGroup (subclass of DoTS::AASequenceGroup)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_SEQUENCE_GROUP_ID	no	NUMBER(12,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
NAME		STRING(500)	Edit
DESCRIPTION		STRING(2000)	Edit
NUMBER_OF_MEMBERS	no	NUMBER(12,0)	Edit
MAX_MATCH_IDENTITY		FLOAT	Edit
MIN_MATCH_IDENTITY		FLOAT	Edit
MAX_MATCH_LENGTH		NUMBER(12,0)	Edit
MIN_MATCH_LENGTH		NUMBER(12,0)	Edit
MAX_PVALUE_MANT		FLOAT	Edit
MAX_PVALUE_EXP		NUMBER(12,0)	Edit
MIN_PVALUE_MANT		FLOAT	Edit
MIN_PVALUE_EXP		NUMBER(12,0)	Edit
MAX_SCORE		FLOAT	Edit
MIN_SCORE		FLOAT	Edit
AA_SEQ_GROUP_EXPERIMENT_ID		DoTS::AASeqGroupExperiment (NUMBER(12,0)) Edit	

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::AASequenceGroup

column	nulls?	type	description
AA_SEQUENCE_GROUP_ID	no	NUMBER(12,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
NAME		STRING(500)	Edit
DESCRIPTION		STRING(2000)	Edit
NUMBER_OF_MEMBERS	no	NUMBER(12,0)	Edit
MAX_MATCH_IDENTITY		FLOAT	Edit
MIN_MATCH_IDENTITY		FLOAT	Edit
MAX_MATCH_LENGTH		NUMBER(12,0)	Edit
MIN_MATCH_LENGTH		NUMBER(12,0)	Edit
MAX_PVALUE_MANT		FLOAT	Edit
MAX_PVALUE_EXP		NUMBER(12,0)	Edit
MIN_PVALUE_MANT		FLOAT	Edit
MIN_PVALUE_EXP		NUMBER(12,0)	Edit
MAX_SCORE		FLOAT	Edit
MIN_SCORE		FLOAT	Edit
AA_SEQ_GROUP_EXPERIMENT_ID		DoTS::AASeqGroupExperiment (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Paralog and Family
 DoTS::AASequenceGroupFamily
AA Ortholog
 DoTS::AASequenceSequenceGroup

Subclasses:

DoTS::AAOrthologGroup
DoTS::AAParalogGroup

---

webmaster@gusdb.org

GUS Schema >> DoTS::AASequenceFamily

column	nulls?	type	description
AA_SEQUENCE_FAMILY_ID	no	NUMBER(12,0)	Edit
NAME	no	STRING(500)	Edit
DESCRIPTION		STRING(2000)	Edit
AA_FAMILY_EXPERIMENT_ID		DoTS::AAFamilyExperiment (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Paralog and Family
  DoTS::AASequenceGroupFamily

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::AASequenceGroupFamily

column	nulls?	type	description
AA_SEQUENCE_GROUP_FAMILY_ID	no	NUMBER(12,0)	Edit
AA_SEQUENCE_FAMILY_ID		DoTS::AASequenceFamily (NUMBER(12,0))	Edit
AA_SEQUENCE_GROUP_ID		DoTS::AASequenceGroup (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::InParalogExperiment (subclass of DoTS::SeqGroupExperiment)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
SEQ_GROUP_EXPERIMENT_ID	no	NUMBER	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
DESCRIPTION		STRING(2000)	Edit
SEQUENCE_SOURCE		STRING(500)	Edit
PVALUE_MANT		FLOAT	Edit
PVALUE_EXP		NUMBER	Edit
PERCENT_IDENTITY		FLOAT	Edit
PERCENT_MATCH		FLOAT	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

[GUS Schema](#) >> DoTS::SeqGroupExperiment

column	nulls?	type	description
SEQ_GROUP_EXPERIMENT_ID	no	NUMBER	[Edit](#)
SUBCLASS_VIEW	no	STRING(30)	[Edit](#)
DESCRIPTION		STRING(2000)	[Edit](#)
SEQUENCE_SOURCE		STRING(500)	[Edit](#)
PVALUE_MANT		FLOAT	[Edit](#)
PVALUE_EXP		NUMBER	[Edit](#)
PERCENT_IDENTITY		FLOAT	[Edit](#)
PERCENT_MATCH		FLOAT	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

Paralog and Family
    [DoTS::SequenceFamilyExperiment](#)
    [DoTS::SequenceGroup](#)
AA Ortholog
    [DoTS::BestSimilarityPair](#)

Subclasses:

[DoTS::OrthologExperiment](#)
[DoTS::OutParalogExperiment](#)
[DoTS::InParalogExperiment](#)

GUS Schema >> DoTS::InParalogGroup (subclass of DoTS::SequenceGroup)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
SEQUENCE_GROUP_ID	no	NUMBER	Edit
SUBCLASS_VIEW	no	STRING(100)	Edit
NAME		STRING(500)	Edit
DESCRIPTION		STRING(2000)	Edit
NUMBER_OF_MEMBERS	no	NUMBER	Edit
NUMBER_OF_TAXA		NUMBER	Edit
MAX_MATCH_IDENTITY		FLOAT	Edit
MIN_MATCH_IDENTITY		FLOAT	Edit
MAX_PERCENT_MATCH		FLOAT	Edit
MIN_PERCENT_MATCH		FLOAT	Edit
MAX_PVALUE_MANT		FLOAT	Edit
MAX_PVALUE_EXP		NUMBER	Edit
MIN_PVALUE_MANT		FLOAT	Edit
MIN_PVALUE_EXP		NUMBER	Edit
MAX_SCORE		FLOAT	Edit
MIN_SCORE		FLOAT	Edit
SEQUENCE_GROUP_EXPERIMENT_ID		DoTS::SeqGroupExperiment (NUMBER)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

[GUS Schema](#) >> DoTS::SequenceGroup

column	nulls?	type	description
SEQUENCE_GROUP_ID	no	NUMBER	Edit
SUBCLASS_VIEW	no	STRING(100)	Edit
NAME		STRING(500)	Edit
DESCRIPTION		STRING(2000)	Edit
NUMBER_OF_MEMBERS	no	NUMBER	Edit
NUMBER_OF_TAXA		NUMBER	Edit
MAX_MATCH_IDENTITY		FLOAT	Edit
MIN_MATCH_IDENTITY		FLOAT	Edit
MAX_PERCENT_MATCH		FLOAT	Edit
MIN_PERCENT_MATCH		FLOAT	Edit
MAX_PVALUE_MANT		FLOAT	Edit
MAX_PVALUE_EXP		NUMBER	Edit
MIN_PVALUE_MANT		FLOAT	Edit
MIN_PVALUE_EXP		NUMBER	Edit
MAX_SCORE		FLOAT	Edit
MIN_SCORE		FLOAT	Edit
SEQUENCE_GROUP_EXPERIMENT_ID		DoTS::SeqGroupExperiment (NUMBER)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Paralog and Family
   DoTS::SequenceGroupFamily
   DoTS::SequenceSequenceGroup

Subclasses:

DoTS::InParalogGroup
DoTS::OrthologGroup
DoTS::OutParalogGroup

webmaster@gusdb.org

GUS Schema >> DoTS::OrthologExperiment (subclass of DoTS::SeqGroupExperiment)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
SEQ_GROUP_EXPERIMENT_ID	no	NUMBER	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
DESCRIPTION		STRING(2000)	Edit
SEQUENCE_SOURCE		STRING(500)	Edit
PVALUE_MANT		FLOAT	Edit
PVALUE_EXP		NUMBER	Edit
PERCENT_IDENTITY		FLOAT	Edit
PERCENT_MATCH		FLOAT	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::OrthologGroup (subclass of DoTS::SequenceGroup)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
SEQUENCE_GROUP_ID	no	NUMBER	Edit
SUBCLASS_VIEW	no	STRING(100)	Edit
NAME		STRING(500)	Edit
DESCRIPTION		STRING(2000)	Edit
NUMBER_OF_MEMBERS	no	NUMBER	Edit
NUMBER_OF_TAXA		NUMBER	Edit
MAX_MATCH_IDENTITY		FLOAT	Edit
MIN_MATCH_IDENTITY		FLOAT	Edit
MAX_PERCENT_MATCH		FLOAT	Edit
MIN_PERCENT_MATCH		FLOAT	Edit
MAX_PVALUE_MANT		FLOAT	Edit
MAX_PVALUE_EXP		NUMBER	Edit
MIN_PVALUE_MANT		FLOAT	Edit
MIN_PVALUE_EXP		NUMBER	Edit
MAX_SCORE		FLOAT	Edit
MIN_SCORE		FLOAT	Edit
SEQUENCE_GROUP_EXPERIMENT_ID		DoTS::SeqGroupExperiment (NUMBER)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::OutParalogExperiment (subclass of DoTS::SeqGroupExperiment)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
SEQ_GROUP_EXPERIMENT_ID	no	NUMBER	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
DESCRIPTION		STRING(2000)	Edit
SEQUENCE_SOURCE		STRING(500)	Edit
PVALUE_MANT		FLOAT	Edit
PVALUE_EXP		NUMBER	Edit
PERCENT_IDENTITY		FLOAT	Edit
PERCENT_MATCH		FLOAT	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::OutParalogGroup (subclass of DoTS::SequenceGroup)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
SEQUENCE_GROUP_ID	no	NUMBER	Edit
SUBCLASS_VIEW	no	STRING(100)	Edit
NAME		STRING(500)	Edit
DESCRIPTION		STRING(2000)	Edit
NUMBER_OF_MEMBERS	no	NUMBER	Edit
NUMBER_OF_TAXA		NUMBER	Edit
MAX_MATCH_IDENTITY		FLOAT	Edit
MIN_MATCH_IDENTITY		FLOAT	Edit
MAX_PERCENT_MATCH		FLOAT	Edit
MIN_PERCENT_MATCH		FLOAT	Edit
MAX_PVALUE_MANT		FLOAT	Edit
MAX_PVALUE_EXP		NUMBER	Edit
MIN_PVALUE_MANT		FLOAT	Edit
MIN_PVALUE_EXP		NUMBER	Edit
MAX_SCORE		FLOAT	Edit
MIN_SCORE		FLOAT	Edit
SEQUENCE_GROUP_EXPERIMENT_ID		DoTS::SeqGroupExperiment (NUMBER)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::SequenceFamily

column	nulls?	type	description
SEQUENCE_FAMILY_ID	no	NUMBER	Edit
NAME	no	STRING(500)	Edit
DESCRIPTION		STRING(2000)	Edit
SEQUENCE_FAMILY_EXPERIMENT_ID		DoTS::SequenceFamilyExperiment (NUMBER)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Paralog and Family
   DoTS::SequenceGroupFamily

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::SequenceFamilyExperiment

column	nulls?	type	description
SEQUENCE_FAMILY_EXPERIMENT_ID	no	NUMBER	Edit
DESCRIPTION		STRING(2000)	Edit
ORTHOLOG_EXPERIMENT_ID		DoTS::SeqGroupExperiment (NUMBER)	Edit
IN_PARALOG_EXPERIMENT_ID		DoTS::SeqGroupExperiment (NUMBER)	Edit
OUT_PARALOG_EXPERIMENT_ID		DoTS::SeqGroupExperiment (NUMBER)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Paralog and Family
> DoTS::SequenceFamily

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::SequenceGroupFamily

column	nulls?	type	description
SEQUENCE_GROUP_FAMILY_ID	no	NUMBER	Edit
SEQUENCE_FAMILY_ID		DoTS::SequenceFamily (NUMBER)	Edit
SEQUENCE_GROUP_ID		DoTS::SequenceGroup (NUMBER)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::SequenceSequenceGroup

column	nulls?	type	description
SEQUENCE_SEQUENCE_GROUP_ID	no	NUMBER	Edit
SEQUENCE_ID	no	NUMBER	Edit
SOURCE_TABLE_ID		Core::TableInfo (NUMBER(5,0))	Edit
SEQUENCE_GROUP_ID	no	DoTS::SequenceGroup (NUMBER)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::AAOrthologExperiment (subclass of DoTS::AASeqGroupExperiment)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_SEQ_GROUP_EXPERIMENT_ID	no	NUMBER(12,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
DESCRIPTION		STRING(2000)	Edit
SEQUENCE_SOURCE		STRING(500)	Edit
PVALUE_MANT		FLOAT	Edit
PVALUE_EXP		NUMBER(12,0)	Edit
PERCENT_IDENTITY		FLOAT	Edit
PERCENT_MATCH		FLOAT	Edit

Child tables:

Subclasses:

GUS Schema >> DoTS::AAOrthologGroup (subclass of DoTS::AASequenceGroup)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
AA_SEQUENCE_GROUP_ID	no	NUMBER(12,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
NAME		STRING(500)	Edit
DESCRIPTION		STRING(2000)	Edit
NUMBER_OF_MEMBERS	no	NUMBER(12,0)	Edit
MAX_MATCH_IDENTITY		FLOAT	Edit
MIN_MATCH_IDENTITY		FLOAT	Edit
MAX_MATCH_LENGTH		NUMBER(12,0)	Edit
MIN_MATCH_LENGTH		NUMBER(12,0)	Edit
MAX_PVALUE_MANT		FLOAT	Edit
MAX_PVALUE_EXP		NUMBER(12,0)	Edit
MIN_PVALUE_MANT		FLOAT	Edit
MIN_PVALUE_EXP		NUMBER(12,0)	Edit
MAX_SCORE		FLOAT	Edit
MIN_SCORE		FLOAT	Edit
AA_SEQ_GROUP_EXPERIMENT_ID		DoTS::AASeqGroupExperiment (NUMBER(12,0))	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

gusdb.org | Schema Browser

GUS Schema >> DoTS::AASequenceSequenceGroup

column	nulls?	type	description
AA_SEQUENCE_SEQUENCE_GROUP_ID	no	NUMBER(12,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(12,0))	Edit
AA_SEQUENCE_GROUP_ID	no	DoTS::AASequenceGroup (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

http://www.gusdb.org/SchemaBrowser/table.htm?schema=DoTS&table=AASequenceSequenceGroup[2/26/13 3:59:31 PM]

gusdb.org | Schema Browser

GUS Schema >> DoTS::BestSimilarityPair

column	nulls?	type	description
BEST_SIMILARITY_PAIR_ID	no	NUMBER	Edit
SEQUENCE_ID	no	NUMBER	Edit
PAIRED_SEQUENCE_ID	no	NUMBER	Edit
SOURCE_TABLE_ID		Core::TableInfo (NUMBER(5,0))	Edit
PAIRED_SOURCE_TABLE_ID		Core::TableInfo (NUMBER(5,0))	Edit
PVALUE_EXP		NUMBER	Edit
PVALUE_MANT		FLOAT	Edit
SCORE		FLOAT	Edit
PERCENT_IDENTITY		FLOAT	Edit
PERCENT_MATCH		FLOAT	Edit
ORTHOLOG_EXPERIMENT_ID		DoTS::SeqGroupExperiment (NUMBER)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::EndSequencePairMap

column	nulls?	type	description
END_SEQUENCE_PAIR_MAP_ID	no	NUMBER(12,0)	Edit
EXTERNAL_DATABASE_RELEASE_ID	no	SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID	no	STRING(16)	Edit
SECONDARY_ID		STRING(300)	Edit
DESCRIPTION		STRING(300)	Edit
TAXON_ID		SRes::Taxon (NUMBER(12,0))	Edit
NA_SEQUENCE_ID_1	no	DoTS::NASequence (NUMBER(12,0))	Edit
NA_SEQUENCE_ID_2	no	DoTS::NASequence (NUMBER(12,0))	Edit
MIN_PRODUCT_LENGTH		NUMBER	Edit
MAX_PRODUCT_LENGTH		NUMBER	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Misc Applications
   DoTS::GeneFeatureSeqOverlap

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::EPCR

column	nulls?	type	description
EPCR_ID	no	NUMBER(12,0)	Edit
MAP_TABLE_ID	no	Core::TableInfo (NUMBER(5,0))	Edit
MAP_ID	no	NUMBER(12,0)	Edit
NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(10,0))	Edit
SEQ_SUBCLASS_VIEW	no	STRING(30)	Edit
START_POS	no	NUMBER(12,0)	Edit
STOP_POS	no	NUMBER(12,0)	Edit
IS_REVERSED		NUMBER(1,0)	Edit
SIMILARITY_ID_1		NUMBER(12,0)	Edit
SIMILARITY_ID_2		NUMBER(12,0)	Edit
SCORE		FLOAT	Edit
NUM_MISMATCHES		NUMBER(1,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Misc Applications
   DoTS::GeneFeatureSeqOverlap

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> DoTS::FingerprintClone

column	nulls?	type	description
FINGERPRINT_CLONE_ID	no	NUMBER(12,0)	Edit
SOURCE_ID	no	STRING(30)	Edit
EXTERNAL_DATABASE_RELEASE_ID	no	[SRes::ExternalDatabaseRelease](#) (NUMBER(10,0))	Edit
IS_SEQUENCED	no	STRING(1)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Mapping
[DoTS::FingerprintCloneContig](#)
[DoTS::FingerprintCloneMarker](#)

Subclasses:

[webmaster@gusdb.org](#)

GUS Schema >> DoTS::FingerprintCloneContig

column	nulls?	type	description
FINGERPRINT_CLONE_CONTIG_ID	no	NUMBER(12,0)	Edit
FINGERPRINT_CLONE_ID	no	DoTS::FingerprintClone (NUMBER(12,0))	Edit
FINGERPRINT_CONTIG_ID	no	DoTS::FingerprintContig (NUMBER(12,0))	Edit
CLONE_ORDER_NUM	no	NUMBER(8,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> DoTS::FingerprintCloneMarker

column	nulls?	type	description
FINGERPRINT_CLONE_MARKER_ID	no	NUMBER(12,0)	[Edit](#)
FINGERPRINT_MAP_ID	no	[DoTS::FingerprintMap](#) (NUMBER(12,0))	[Edit](#)
FINGERPRINT_CLONE_ID	no	[DoTS::FingerprintClone](#) (NUMBER(12,0))	[Edit](#)
MARKER_TYPE		STRING(30)	[Edit](#)
SOURCE_ID	no	STRING(30)	[Edit](#)
EXTERNAL_DATABASE_RELEASE_ID		[SRes::ExternalDatabaseRelease](#) (NUMBER(10,0))	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

Subclasses:

[webmaster@gusdb.org](#)

GUS Schema >> DoTS::FingerprintContig

column	nulls?	type	description
FINGERPRINT_CONTIG_ID	no	NUMBER(12,0)	Edit
FINGERPRINT_MAP_ID	no	DoTS::FingerprintMap (NUMBER(12,0))	Edit
NAME	no	STRING(30)	Edit
NUMBER_OF_CLONES	no	NUMBER(10,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Mapping
   DoTS::FingerprintCloneContig

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> DoTS::FingerprintMap

column	nulls?	type	description
FINGERPRINT_MAP_ID	no	NUMBER(12,0)	[Edit](#)
NAME	no	STRING(80)	[Edit](#)
MAP_VERSION	no	STRING(30)	[Edit](#)
EXTERNAL_DATABASE_RELEASE_ID	no	[SRes::ExternalDatabaseRelease](#) (NUMBER(10,0))	[Edit](#)
SOURCE_ID		STRING(32)	[Edit](#)
NUMBER_OF_CLONES		NUMBER(10,0)	[Edit](#)
NUMBER_OF_CONTIGS		NUMBER(10,0)	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

Mapping
    [DoTS::FingerprintCloneMarker](#)
    [DoTS::FingerprintContig](#)

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::MicroSatelliteMap

column	nulls?	type	description
PLASMOMAP_ID	no	NUMBER(12,0)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID	no	STRING(15)	Edit
MARKER_NAME	no	STRING(25)	Edit
MARKER_SYNONYM		STRING(15)	Edit
ACCESSION		STRING(10)	Edit
CHROMOSOME	no	STRING(2)	Edit
CENTIMORGANS		FLOAT	Edit
FRAMEWORK		STRING(15)	Edit
FORWARD_PRIMER	no	STRING(50)	Edit
REVERSE_PRIMER	no	STRING(50)	Edit
PRODUCT_LENGTH	no	NUMBER(12,0)	Edit
TAXON_ID	no	SRes::Taxon (NUMBER(12,0))	Edit
OUTCROSS	no	STRING(30)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::OpticalMapAlignment

column	nulls?	type	description
OPTICAL_MAP_ALIGNMENT_ID	no	NUMBER(12,0)	Edit
OPTICAL_MAP_ID	no	DoTS::OpticalMap (NUMBER(12,0))	Edit
NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(12,0))	Edit
SCORE		FLOAT	Edit
P_VALUE	no	FLOAT	Edit
START_FRAGMENT	no	NUMBER(5,0)	Edit
END_FRAGMENT	no	NUMBER(5,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Mapping
DoTS::OpticalMapAlignmentSpan

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> **DoTS::OpticalMapAlignmentSpan**

column	nulls?	type	description
OPTICAL_MAP_ALIGNMENT_SPAN_ID	no	NUMBER(12,0)	[Edit](#)
OPTICAL_MAP_ALIGNMENT_ID	no	[DoTS::OpticalMapAlignment](#) (NUMBER(12,0))	[Edit](#)
OPTICAL_MAP_FRAGMENT_ID		[DoTS::OpticalMapFragment](#) (NUMBER(12,0))	[Edit](#)
NA_Feature_ID		[DoTS::NAFeature](#) (NUMBER(12,0))	[Edit](#)
SCORE		FLOAT	[Edit](#)
STATE_NAME		STRING(25)	[Edit](#)
OPTICAL_MAP_LENGTH		FLOAT	[Edit](#)
FRAGMENT_LENGTH_KB		FLOAT	[Edit](#)
ORDER_NUM		NUMBER(5,0)	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::OpticalMapFragment

column	nulls?	type	description
OPTICAL_MAP_FRAGMENT_ID	no	NUMBER(12,0)	Edit
OPTICAL_MAP_ID	no	DoTS::OpticalMap (NUMBER(12,0))	Edit
FRAGMENT_LENGTH_KB	no	FLOAT	Edit
FRAGMENT_ORDER	no	NUMBER(5,0)	Edit
CUMULATIVE_LENGTH_KB		FLOAT	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Mapping
DoTS::OpticalMapAlignmentSpan

Subclasses:

GUS Schema >> DoTS::RHMap

column	nulls?	type	description
RH_MAP_ID	no	NUMBER(12,0)	Edit
NAME	no	STRING(100)	Edit
TAXON_ID	no	SRes::Taxon (NUMBER(12,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	Edit
SOURCE_ID		STRING(32)	Edit
LAB		STRING(200)	Edit
DESCRIPTION		STRING(4000)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Mapping
    DoTS::RHMapMarker

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::RHMapMarker

column	nulls?	type	description
RH_MAP_MARKER_ID	no	NUMBER(12,0)	Edit
RH_MARKER_ID	no	DoTS::RHMarker (NUMBER(12,0))	Edit
RH_MAP_ID	no	DoTS::RHMap (NUMBER(12,0))	Edit
CHROMOSOME		STRING(8)	Edit
CENTIRAYS	no	FLOAT	Edit
CENTIMORGANS		FLOAT	Edit
FRAME_PLACE		STRING(2)	Edit
LOD		STRING(10)	Edit
RH_VECTOR		STRING(255)	Edit
LAB		STRING(8)	Edit
ASSAY		STRING(200)	Edit
PANEL_DESCRIPTION		STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::RHMarker

column	nulls?	type	description
RH_MARKER_ID	no	NUMBER(12,0)	Edit
EXTERNAL_DATABASE_RELEASE_ID	no	SRes::ExternalDatabaseRelease (NUMBER(12,0))	Edit
SOURCE_ID		STRING(30)	Edit
NAME	no	STRING(200)	Edit
PANEL		STRING(20)	Edit
GENBANK_ID		STRING(15)	Edit
FORWARD_PRIMER		STRING(50)	Edit
REVERSE_PRIMER		STRING(50)	Edit
PRODUCT_LENGTH		NUMBER(12,0)	Edit
MARKER_SEQUENCE		CLOB	Edit
DESCRIPTION		STRING(500)	Edit
TAXON_ID	no	SRes::Taxon (NUMBER(12,0))	Edit
TYPE		STRING(8)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Mapping
   DoTS::RHMapMarker

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::AnatomyLibrary

Table containing the anatomy_id from the anatomy controlled vocabulary that has been assigned to a library from dbest using its dbest_library_id

column	nulls?	type	description
ANATOMY_LIBRARY_ID	no	NUMBER(10,0)	primary key Edit
ANATOMY_ID	no	SRes::Anatomy (NUMBER(4,0))	The anatomy term identifier Edit
DBEST_LIBRARY_ID	no	NUMBER(12,0)	The dbest library identifer Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::AnatomyLOE

A controlled vocabulary describing the experiment type used as a line of evidence in the RNAAnatomy assignment

column	nulls?	type	description
ANATOMY_LOE_ID	no	NUMBER(5,0)	primary key Edit
NAME	no	STRING(255)	"experiment type (e.g., microarray)" Edit
DESCRIPTION		STRING(500)	a definition of the type of experiment performed Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Central Dogma
   DoTS::RNAAnatomyLOE

Subclasses:

---

webmaster@gusdb.org

gusdb.org | Schema Browser

GUS Schema >> DoTS::Clone

Tracks clone information (e.g. IMAGE clone id)

column	nulls?	type	description
CLONE_ID	no	NUMBER(10,0)	Primary key Edit
LIBRARY_ID	no	DoTS::Library (NUMBER(5,0))	Foreign key to Library table Edit
IMAGE_ID		NUMBER(12,0)	IMAGE clone id (just the integer bit) Edit
DBEST_CLONE_UID		STRING(64)	Parsed string from dbEST e_uid. Not gaurenteed to be unique or useful (such as an IMAGE id) Edit
WASHU_NAME		STRING(64)	The name of the clone according to the WashU sequencing center Edit
GDB_ID		NUMBER(12,0)	GDB identifier Edit
MGI_ID		NUMBER(12,0)	MGI identifier Edit
DBEST_LENGTH		NUMBER(12,0)	Parse value from dbEST EST.length Edit
WASHU_LENGTH		NUMBER(12,0)	The length of the clone insert according to WashU sequencing center Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Clones
   DoTS::CloneInSet
   DoTS::EST

Subclasses:

[GUS Schema](#) >> **DoTS::CloneInSet**

Relates entries in the Clone table to some CloneSet (e.g. the EPConDB clone set 1.0)

column	nulls?	type	description
CLONE_IN_SET_ID	no	NUMBER(10,0)	Primary key Edit
CLONE_ID	no	DoTS::Clone (NUMBER(10,0))	FK to Clone Edit
CLONE_SET_ID	no	DoTS::CloneSet (NUMBER(12,0))	FK to CloneSet Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::CloneSet

Describes a set of clones (e.g. the EPConDB clone set 1.0)

column	nulls?	type	description
CLONE_SET_ID	no	NUMBER(12,0)	PK Edit
CONTACT_ID	no	SRes::Contact (NUMBER(12,0))	FK to Contact Edit
NAME	no	STRING(20)	the name of the clone set Edit
EXTERNAL_NAME		STRING(30)	Alternate name Edit
DESCRIPTION		STRING(255)	Describes the clone set Edit
CBIL_NAME		STRING(36)	Alternate name specific to CBIL Edit
DATE_IN		DATE	Date the clone set was entered into GUS Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Clones
   DoTS::CloneInSet

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::DevStageLibrary

column	nulls?	type	description
DEV_STAGE_LIBRARY_ID	no	NUMBER(10,0)	Edit
DEVELOPMENTAL_STAGE_ID	no	SRes::DevelopmentalStage (NUMBER(4,0))	Edit
LIBRARY_ID	no	DoTS::Library (NUMBER(5,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

GUS Schema >> DoTS::EST

column	nulls?	type	description
EST_ID	no	NUMBER(10,0)	Edit
NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(10,0))	Edit
CLONE_ID		DoTS::Clone (NUMBER(10,0))	Edit
LIBRARY_ID	no	DoTS::Library (NUMBER(5,0))	Edit
CONTACT_ID	no	SRes::Contact (NUMBER(12,0))	Edit
DBEST_ID_EST		NUMBER(12,0)	Edit
WASHU_ID		STRING(67)	Edit
ACCESSION		STRING(20)	Edit
P_END		CHARACTER(1)	Edit
QUALITY_START	no	NUMBER(12,0)	Edit
QUALITY_STOP		NUMBER(12,0)	Edit
SEQ_LENGTH		NUMBER(12,0)	Edit
SEQ_PRIMER		STRING(128)	Edit
POSSIBLY_REVERSED	no	NUMBER(1,0)	Edit
PUTATIVE_FULL_LENGTH_READ	no	NUMBER(1,0)	Edit
TRACE_POOR_QUALITY	no	NUMBER(1,0)	Edit
POLYA_SIGNAL		NUMBER(1,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

gusdb.org | Schema Browser

GUS Schema >> DoTS::Library

Stores clone library information

column	nulls?	type	description
LIBRARY_ID	no	NUMBER(5,0)	Primary key column Edit
TAXON_ID	no	SRes::Taxon (NUMBER(12,0))	FK to Taxon (Parsed from dbEST.LIBRARY.organism) Edit
ANATOMY_ID		SRes::Anatomy (NUMBER(4,0))	FK to Anatomy (Parsed from dbEST.LIBRARY.organ) Edit
DBEST_ID		NUMBER(12,0)	The dbEST library id (dbEST.LIBRARY.id_lib) Edit
DBEST_NAME		STRING(120)	The name according to dbEST (dbEST.LIBRARY.name) Edit
IS_IMAGE	no	NUMBER(1,0)	Whether this is an IMAGE library Edit
IMAGE_ID		NUMBER(12,0)	The IMAGE Consortium library id Edit
GDB_ID		NUMBER(12,0)	GDB library id Edit
GDB_NAME		STRING(100)	The name according to GDB Edit
LLNL_NAME		STRING(100)	The name according to LLNL Edit
WASHU_NAME		STRING(100)	The name according to WashU Edit
TISSUE_TYPE		STRING(156)	The tissue type of the library (dbEST.LIBRARY.tissue_type) Edit
STRAIN		STRING(156)	Organism strain (dbEST.LIBRARY.strain) Edit
CULTIVAR		STRING(156)	Plant cultivar (dbEST.LIBRARY.cultivar) Edit
STAGE		STRING(100)	Developmental stage (dbEST.LIBRARY.stage) Edit
SEX		CHARACTER(1)	Sex of organism (female, male, hermaphrodite) (parsed from dbEST.LIBRARY.sex) (values are currently = [S,-,?,B,M,F,n]) Edit
CELL_TYPE		STRING(156)	Cell type (dbEST.LIBRARY.cell_type) Edit
CELL_LINE		STRING(156)	Name of cell line (dbEST.LIBRARY.cell_line) Edit
NORMALIZED		CHARACTER(1)	Whether the library was normalized Edit
VECTOR		STRING(156)	Name of the vector (dbEST.LIBRARY.vector) Edit
VECTOR_TYPE		STRING(64)	Type of the vector (dbEST.LIBRARY.v_type) Edit
HOST		STRING(80)	Name of the host Edit
RE_1		STRING(35)	Restriction enzyme at site1 of vector (dbEST.LIBRARY.re_1) Edit
RE_2		STRING(35)	Restriction enzyme at site2 of vector (dbEST.LIBRARY.re_2) Edit
COMMENT_STRING		CLOB	Edit
DBEST_ORGANISM		STRING(300)	We store dbEST.LIBRARY.organism for debugging Edit
DBEST_ORGAN		STRING(300)	We store dbEST.LIBRARY.organ for debugging Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Clones
   DoTS::Clone
   DoTS::DevStageLibrary
   DoTS::EST

Subclasses:

gusdb.org | Schema Browser

GUS Schema >> DoTS::Complex

Table used to specify a complex. A complex consists of many ComplexComponents

column	nulls?	type	description
COMPLEX_ID	no	NUMBER(10,0)	Primary key, used to uniquely identify a complex Edit
NAME		STRING(80)	Name of the complex Edit
DESCRIPTION		STRING(255)	Description of the complex Edit
COMPLEX_TYPE_ID	no	DoTS::ComplexType (NUMBER(10,0))	Foreign key to ComplexType; specifies the type of the complex Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Foreign key to SRes::ExternalDatabaseRelease; this attribute is used when the complex has been loaded from an external source Edit
SOURCE_ID		STRING(32)	The primary identifier of the complex in the external database specified by external_database_release_id Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Interaction
 DoTS::ComplexAnatomy
 DoTS::ComplexComponent

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::ComplexAnatomy

Many to many table used to relate Complex to Anatomy

column	nulls?	type	description
COMPLEX_ANATOMY_ID	no	NUMBER(10,0)	Primary key, used to uniquely identify a ComplexAnatomy Edit
COMPLEX_ID	no	DoTS::Complex (NUMBER(10,0))	Foreign key to Complex, specifies the complex that is associated with the anatomy_id Edit
ANATOMY_ID	no	SRes::Anatomy (NUMBER(4,0))	Foreign key to SRes::Anatomy, specifies an anatomy where the complex is found Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::ComplexComponent

A many to many table used to relate any row in the database to a complex. The ComplexComponent then becomes part of the Complex

column	nulls?	type	description
COMPLEX_COMPONENT_ID	no	NUMBER(10,0)	Primary key, used to uniquely identify a ComplexComponent Edit
TABLE_ID	no	Core::TableInfo (NUMBER(5,0))	Foreign key to CORE::TableInfo, specifies the table id of the component of the complex specified by complex_id Edit
ROW_ID	no	NUMBER(10,0)	The single attribute primary key of the row in the table specified by table_id that is a member of the complex specified by complex_id Edit
COMPLEX_ID	no	DoTS::Complex (NUMBER(10,0))	Foreign key to Complex; specifies which complex the component belongs to Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::ComplexType

column	nulls?	type	description
COMPLEX_TYPE_ID	no	NUMBER(5,0)	Edit
NAME	no	STRING(80)	Edit
DESCRIPTION		STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Interaction
   DoTS::Complex

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::EffectorActionType

column	nulls?	type	description
EFFECTOR_Action_TYPE_ID	no	NUMBER(5,0)	Edit
NAME	no	STRING(80)	Edit
DESCRIPTION		STRING(255)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(32)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Interaction
DoTS::Interaction

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::Interaction

Table used to hold interactions

column	nulls?	type	description
INTERAction_ID	no	NUMBER(10,0)	Primary key, used to uniquely identify an interaction Edit
INTERAction_TYPE_ID	no	DoTS::InteractionType (NUMBER(5,0))	Foreign key to InteractionType, specifies the type of interaction Edit
EFFECTOR_Action_TYPE_ID	no	DoTS::EffectorActionType (NUMBER(5,0))	Edit
EFFECTOR_ROW_SET_ID	no	DoTS::RowSet (NUMBER(10,0))	Edit
TARGET_ROW_SET_ID	no	DoTS::RowSet (NUMBER(10,0))	Edit
HAS_DIRECTION	no	NUMBER(1,0)	Edit
DIRECTION_IS_KNOWN	no	NUMBER(1,0)	Set to 1 if the direction of the interaction is known; 0 otherwise. Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Foreign key to SRes::ExternalDatabaseRelease; this attribute is used when the interaction has been loaded from an external source Edit
SOURCE_ID		STRING(32)	The primary identifier of the interaction in the external database specified by external_db_release_id Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Interaction
    DoTS::InteractionInteractionLOE
    DoTS::PathwayInteraction

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::InteractionInteractionLOE

Table to associate the line of evidence with the Interaction assignments

column	nulls?	type	description
INTERAction_INTERAction_LOE_ID	no	NUMBER(10,0)	primary key Edit
INTERAction_ID	no	DoTS::Interaction (NUMBER(10,0))	foreign key to interaction Edit
INTERAction_LOE_ID	no	DoTS::InteractionLOE (NUMBER(10,0))	foreign key to interactionLOE Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

GUS Schema >> DoTS::InteractionLOE

A controlled vocabulary describing the experiment type used as a line of evidence in the interaction assignment

column	nulls?	type	description
INTERAction_LOE_ID	no	NUMBER(10,0)	primary key Edit
NAME	no	STRING(80)	"experiment type (e.g., yeast 2-hybrid)" Edit
DESCRIPTION		STRING(255)	a definition of the type of experiment performed Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Interaction
   DoTS::InteractionInteractionLOE

Subclasses:

GUS Schema >> DoTS::InteractionType

column	nulls?	type	description
INTERAction_TYPE_ID	no	NUMBER(5,0)	Edit
NAME	no	STRING(80)	Edit
DESCRIPTION		STRING(255)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(32)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Interaction
DoTS::Interaction

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::Pathway

Table used to hold pathway information. A pathway is defined by its interactions through the use of the PathwayInteraction relation table

column	nulls?	type	description
PATHWAY_ID	no	NUMBER(10,0)	Primary key, used to uniquely identify a pathway Edit
NAME	no	STRING(80)	A name for the pathway Edit
DESCRIPTION		STRING(255)	A description of the pathway Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Foreign key to SRes::ExternalDatabaseRelease; this attribute is used when the pathway has been loaded from an external source Edit
SOURCE_ID		STRING(32)	The primary identifier of the pathway in the external database specified by external_database_release_id Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Interaction
DoTS::PathwayInteraction

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::PathwayInteraction

Many to Many table used to relate Interactions and Pathways

column	nulls?	type	description
PATHWAY_INTERAction_ID	no	NUMBER(10,0)	Primary key, used to uniquely identify an interaction Edit
PATHWAY_ID	no	DoTS::Pathway (NUMBER(10,0))	Foreign key to Pathway, the interaction specified by interaction_id will become part of this pathway Edit
INTERAction_ID	no	DoTS::Interaction (NUMBER(10,0))	Foreign key to Interaction, specifies the interaction that is part of the pathway specified by pathway_id Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::RowSet

column	nulls?	type	description
ROW_SET_ID	no	NUMBER(10,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Interaction
DoTS::Interaction
DoTS::RowSetMember

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::RowSetMember

column	nulls?	type	description
ROW_SET_MEMBER_ID	no	NUMBER(10,0)	Edit
ROW_SET_ID	no	DoTS::RowSet (NUMBER(10,0))	Edit
TABLE_ID	no	Core::TableInfo (NUMBER(5,0))	Edit
ROW_ID	no	NUMBER(10,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> DoTS::GOAssociation

column	nulls?	type	description
GO_ASSOCIATION_ID	no	NUMBER(10,0)	[Edit](#)
TABLE_ID	no	[Core::TableInfo](#) (NUMBER(10,0))	[Edit](#)
ROW_ID	no	NUMBER(10,0)	[Edit](#)
GO_TERM_ID	no	[SRes::GOTerm](#) (NUMBER(10,0))	[Edit](#)
IS_NOT	no	NUMBER(1,0)	[Edit](#)
IS_DEPRECATED	no	NUMBER(1,0)	[Edit](#)
DEFINING	no	NUMBER(1,0)	[Edit](#)
GO_ASSOCIATION_DATE		DATE	[Edit](#)
REVIEW_STATUS_ID		[SRes::ReviewStatus](#) (NUMBER(12,0))	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

GO Association
[DoTS::GOAssociationInstance](#)

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::GOAssociationInstance

column	nulls?	type	description
GO_ASSOCIATION_INSTANCE_ID	no	NUMBER(10,0)	Edit
GO_ASSOCIATION_ID	no	DoTS::GOAssociation (NUMBER(10,0))	Edit
GO_ASSOC_INST_LOE_ID	no	DoTS::GOAssociationInstanceLOE (NUMBER(5,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(32)	Edit
IS_NOT		NUMBER(1,0)	Edit
IS_PRIMARY	no	NUMBER(1,0)	Edit
IS_DEPRECATED	no	NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
P_VALUE_RATIO		FLOAT	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

GO Association
   DoTS::GOAssocInstEvidCode

Subclasses:

GUS Schema >> DoTS::GOAssociationInstanceLOE

column	nulls?	type	description
GO_ASSOC_INST_LOE_ID	no	NUMBER(5,0)	Edit
NAME	no	STRING(80)	Edit
DESCRIPTION		STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

GO Association
   DoTS::GOAssociationInstance

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::GOAssocInstEvidCode

column	nulls?	type	description
GO_ASSOC_INST_EVID_CODE_ID	no	NUMBER(10,0)	Edit
GO_EVIDENCE_CODE_ID	no	SRes::GOEvidenceCode (NUMBER(3,0))	Edit
GO_ASSOCIATION_INSTANCE_ID	no	DoTS::GOAssociationInstance (NUMBER(10,0))	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::Allele

column	nulls?	type	description
ALLELE_ID	no	NUMBER(10,0)	Edit
NAME	no	STRING(100)	Edit
GENE_ID	no	DoTS::Gene (NUMBER(10,0))	Edit
IS_WILDTYPE		NUMBER(1,0)	Edit
MUTAGEN_ID		SRes::Mutagen (NUMBER(10,0))	Edit
MUTAGENESIS_DESCRIPTION		STRING(1000)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Variation
DoTS::AlleleComplementation
DoTS::AlleleInstance
DoTS::AllelePhenotype
DoTS::AllelePhenotypeClass

Subclasses:

GUS Schema >> DoTS::AlleleComplementation

column	nulls?	type	description
ALLELE_COMPLEMENTATION_ID	no	NUMBER(10,0)	Edit
ALLELE_ID	no	DoTS::Allele (NUMBER(10,0))	Edit
COMPLEMENTATION_ID	no	DoTS::Complementation (NUMBER(10,0))	Edit
TYPE	no	STRING(40)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::AlleleInstance

column	nulls?	type	description
ALLELE_INSTANCE_ID	no	NUMBER(10,0)	Edit
ALLELE_ID	no	DoTS::Allele (NUMBER(10,0))	Edit
NA_Feature_ID	no	DoTS::NAFeature (NUMBER(10,0))	Edit
DESCRIPTION		STRING(500)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> **DoTS::AllelePhenotype**

column	nulls?	type	description
ALLELE_PHENOTYPE_ID	no	NUMBER(10,0)	[Edit](#)
ALLELE_ID	no	[DoTS::Allele](#) (NUMBER(10,0))	[Edit](#)
PHENOTYPE_ID	no	[SRes::Phenotype](#) (NUMBER(10,0))	[Edit](#)
REVIEW_STATUS_ID		[SRes::ReviewStatus](#) (NUMBER(12,0))	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> DoTS::AllelePhenotypeClass

column	nulls?	type	description
ALLELE_PHENOTYPE_CLASS_ID	no	NUMBER(10,0)	[Edit](#)
ALLELE_ID	no	[DoTS::Allele](#) (NUMBER(10,0))	[Edit](#)
PHENOTYPE_CLASS_ID	no	[SRes::PhenotypeClass](#) (NUMBER(10,0))	[Edit](#)
REVIEW_STATUS_ID		[SRes::ReviewStatus](#) (NUMBER(12,0))	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> **DoTS::Complementation**

column	nulls?	type	description
COMPLEMENTATION_ID	no	NUMBER(10,0)	Edit
IS_EXTERNAL_REFERENCE	no	NUMBER(1,0)	Edit
ENTITY_NAME	no	STRING(100)	Edit
TABLE_ID		[Core::TableInfo](#) (NUMBER(10,0))	Edit
ENTRY_ID		NUMBER(10,0)	Edit
TAXON_ID	no	[SRes::Taxon](#) (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		[SRes::ExternalDatabaseRelease](#) (NUMBER(10,0))	Edit
REVIEW_STATUS_ID		[SRes::ReviewStatus](#) (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Variation
[DoTS::AlleleComplementation](#)

Subclasses:

[webmaster@gusdb.org](#)

GUS Schema >> TESS::Moiety

A base class for inheritance.

column	nulls?	type	description
MOIETY_ID	no	NUMBER(12,0)	Primary key. Edit
SUBCLASS_VIEW	no	STRING(32)	Edit
TYPE_NAME	no	STRING(32)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	Edit
SOURCE_ID		STRING(32)	Edit
NAME		STRING(255)	Edit
TAXON_ID	no	SRes::Taxon (NUMBER(12,0))	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Regulatory Moieties
TESS::MoietyInstance
TESS::MoietyMember
TESS::MoietySynonym
Regulatory Activities
TESS::Activity

Subclasses:

TESS::MoietyMultimer
TESS::MoietyHeteroDimer
TESS::MoietyMonomer
TESS::MoietyComplex

GUS Schema >> TESS::MoietyComplex (subclass of TESS::Moiety)

A generic complex class. There can be any number of members in the complex.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
MOIETY_ID	no	NUMBER(12,0)	Primary key. Edit
SUBCLASS_VIEW	no	STRING(32)	Edit
TYPE_NAME	no	STRING(32)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	Edit
SOURCE_ID		STRING(32)	Edit
NAME		STRING(255)	Edit
TAXON_ID	no	SRes::Taxon (NUMBER(12,0))	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
NUMBER_OF_MEMBERS		NUMBER(12,0)	This stores the number of members in the complex. Number is unknown or not relevant if this is NULL. Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> TESS::MoietyHeteroDimer (subclass of TESS::Moiety)

A heterodimer class. It is assumed that there are only two members of the complex and they are different so should be two `TESS.MoietyMember` rows refering to one of these entries.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
MOIETY_ID	no	NUMBER(12,0)	Primary key. Edit
SUBCLASS_VIEW	no	STRING(32)	Edit
TYPE_NAME	no	STRING(32)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	Edit
SOURCE_ID		STRING(32)	Edit
NAME		STRING(255)	Edit
TAXON_ID	no	SRes::Taxon (NUMBER(12,0))	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::MoietyInstance

Moiety objects may be linked to central dogma objects if their identity can be determined.

column	nulls?	type	description
MOIETY_INSTANCE_ID	no	NUMBER(12,0)	Primary key. Edit
MOIETY_ID	no	TESS::Moiety (NUMBER(12,0))	This is the moiety. Edit
DOGMA_OBJECT_TABLE_ID	no	Core::TableInfo (NUMBER(12,0))	This is the central dogma object table. Edit
DOGMA_OBJECT_ID	no	NUMBER(12,0)	This is the ID of the central dogma object. Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	What database release did this link come from. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::MoietyMember

This table allows `TESS.Moiety` to be grouped together to form complexes. There is a whole id which should considered to be the complex, and the child moiety is the member moiety (protein). There should be one `TESS.MoietyMember` row for each member of the complex. A complex can be build up hierarchically or left flat. I guess this might eventually become an Imp table if we start to detail the relationship.

column	nulls?	type	description
MOIETY_MEMBER_ID	no	NUMBER(12,0)	Primary key. Edit
WHOLE_MOIETY_ID	no	TESS::Moiety (NUMBER(12,0))	Link to the parent moiety. Edit
PART_MOIETY_ID	no	TESS::Moiety (NUMBER(12,0))	Link to the parent moiety. Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	What database release did this membership come from. Edit
SOURCE_ID		STRING(32)	What source id did it have in the (external) database. Edit
NAME		STRING(255)	A descriptive name for this instance. Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	What is the review status of this membership. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> TESS::MoietyMonomer (subclass of TESS::Moiety)

A monomer class. It is assumed that members of the complex are the same moiety so there need only be on `TESS.MoietyMember` row refering to one of these entries.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
MOIETY_ID	no	NUMBER(12,0)	Primary key. Edit
SUBCLASS_VIEW	no	STRING(32)	Edit
TYPE_NAME	no	STRING(32)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	Edit
SOURCE_ID		STRING(32)	Edit
NAME		STRING(255)	Edit
TAXON_ID	no	SRes::Taxon (NUMBER(12,0))	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::MoietyMultimer (subclass of TESS::Moiety)

A multimer class. It is assumed that members of the complex are the same moiety so there need only be on `TESS.MoietyMember` row refering to one of these entries.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
MOIETY_ID	no	NUMBER(12,0)	Primary key. Edit
SUBCLASS_VIEW	no	STRING(32)	Edit
TYPE_NAME	no	STRING(32)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	Edit
SOURCE_ID		STRING(32)	Edit
NAME		STRING(255)	Edit
TAXON_ID	no	SRes::Taxon (NUMBER(12,0))	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
MULTIPLICTY		NUMBER(12,0)	This stores the number of monomers in the complex. Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::MoietySynonym

A synonym for a `Moiety` since many factors are known by multiple names.

column	nulls?	type	description
MOIETY_SYNONYM_ID	no	NUMBER(12,0)	Primary key. Edit
MOIETY_ID	no	TESS::Moiety (NUMBER(12,0))	Link to the moiety Edit
SYNONYM_TEXT	no	STRING(255)	The moiety is also referred to by this name or ID. Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	What database (release) did this come from? Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::Activity

This is a base class for inheritance.

column	nulls?	type	description
ACTIVITY_ID	no	NUMBER(12,0)	Primary key. Edit
SUBCLASS_VIEW	no	STRING(32)	Edit
TYPE_NAME	no	STRING(32)	Edit
GO_TERM_ID		SRes::GOTerm (NUMBER(12,0))	Edit
MOIETY_ID		TESS::Moiety (NUMBER(12,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	Edit
SOURCE_ID		STRING(32)	Edit
NAME		STRING(255)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Regulatory Activities
TESS::ActivityConditions
TESS::ActivityInferenceSource
TESS::ActivityModel
Training Sets
TESS::Footprint
TESS::TrainingSet

Subclasses:

TESS::ActivityGeneRegulation
TESS::ActivityProteinDnaBinding

---

webmaster@gusdb.org

GUS Schema >> TESS::ActivityConditions

Storing the conditions in the activity takes place. The links in this table are to be interpreted as logical conjunctions, i.e., all are true simultaneously. If an activity takes place under multiple conditions, e.g., in multiple organs, then each organ would have a record in this table.

column	nulls?	type	description
ACTIVITY_CONDITIONS_ID	no	NUMBER(12,0)	Primary key. Edit
ACTIVITY_ID	no	TESS::Activity (NUMBER(12,0))	A link to the Activity. Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	To what extent has this been reviewed? Edit
TAXON_ID		SRes::Taxon (NUMBER(12,0))	In what species? Edit
ANATOMY_ID		SRes::Anatomy (NUMBER(12,0))	Where in the does 'body' does this activity take place? Edit
CELL_TYPE_ID		SRes::Anatomy (NUMBER(12,0))	In what cell type? Edit
DEVELOPMENTAL_STAGE_ID		SRes::DevelopmentalStage (NUMBER(12,0))	At what developmental stage? Edit
DISEASE_ID		SRes::Disease (NUMBER(12,0))	In what disease state? Edit
PHENOTYPE_ID		SRes::Phenotype (NUMBER(12,0))	In what phenotype? Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::ActivityGeneRegulation (subclass of TESS::Activity)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ACTIVITY_ID	no	NUMBER(12,0)	Primary key. Edit
SUBCLASS_VIEW	no	STRING(32)	Edit
TYPE_NAME	no	STRING(32)	Edit
GO_TERM_ID		SRes::GOTerm (NUMBER(12,0))	Edit
MOIETY_ID		TESS::Moiety (NUMBER(12,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	Edit
SOURCE_ID		STRING(32)	Edit
NAME		STRING(255)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> TESS::ActivityInferenceSource

This table acts as a link between an activity and the result of a gene expression experiment that has identified a group of coordinately regulated genes.

column	nulls?	type	description
ACTIVITY_INFERENCE_SOURCE	no	NUMBER(12,0)	Primary key. Edit
ACTIVITY_ID	no	TESS::Activity (NUMBER(12,0))	The link to the activity. Edit
RESULT_GROUP_ID	no	NUMBER(12,0)	The link the the experimental result group of genes. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::ActivityModel

This table connects a model to an activity. Rows in this table assert that the model describes an arrangement of DNA bases or features that is believed to be at least correlated with the activity taking place at locations in sequence that match the model.

For example, a row in this table might link a DNABinding activity to a weight matrix model of a transcription factor binding site. If a match is found to that weight matrix in genomic sequence, then it may be inferred (with some level of reliability) that the transcription factor will bind there.

Since `TESS.Activity` and `TESS.Model` are views with many subclasses, this table serves to link any kind of model to any kind of activity.

column	nulls?	type	description
ACTIVITY_MODEL_ID	no	NUMBER(12,0)	Primary key Edit
ACTIVITY_ID	no	TESS::Activity (NUMBER(12,0))	This is the gene regulation-related activity the model is supposed to describe. Edit
MODEL_ID	no	TESS::Model (NUMBER(12,0))	This is the model of features of DNA or other sequence that are indicative of the activity. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> TESS::ActivityProteinDnaBinding (subclass of TESS::Activity)

This view of `TESS.ActivityImp` stores DNA–protein interactions between a transcription factor and their DNA binding sites. Each transcription factor will have at least one activity and may have more if it recognizes (distinctly) different sites.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ACTIVITY_ID	no	NUMBER(12,0)	Primary key. Edit
SUBCLASS_VIEW	no	STRING(32)	Edit
TYPE_NAME	no	STRING(32)	Edit
GO_TERM_ID		SRes::GOTerm (NUMBER(12,0))	Edit
MOIETY_ID		TESS::Moiety (NUMBER(12,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	Edit
SOURCE_ID		STRING(32)	Edit
NAME		STRING(255)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

[GUS Schema](#) >> TESS::Note

A place to put (typed) free text for later parsing.

column	nulls?	type	description
NOTE_ID	no	NUMBER(12,0)	Primary key. Edit
TABLE_ID	no	[Core::TableInfo](#) (NUMBER(12,0))	Link to table for this note. Edit
ROW_ID	no	NUMBER(12,0)	ID of the row this note applies to. Edit
NOTE_CLASS	no	STRING(32)	What kind of note is this? Edit
ORDINAL	no	NUMBER	A means of putting notes of a given type in order. Edit
NOTE_TEXT	no	STRING(4000)	The text of the note. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::MarkovChainObs

This table links `TESS::MultinomialObservationSet` to `TESS::ModelMarkovChain` rows.

column	nulls?	type	description
MARKOV_CHAIN_OBS_ID	no	NUMBER(12,0)	Edit
MODEL_ID	no	TESS::Model (NUMBER(12,0))	A link to the Markov Chain model the observation belongs to. Edit
MULTINOMIAL_OBSERVATION_SET_ID	no	TESS::MultinomialObservationSet (NUMBER(12,0))	A link to the observation Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::Model

This table is a many-to-many link between `TESS.Activity` and `TESS.SbcgGrammar`. TRANSFAC will use the same matrix to multiple factors so we are forced to do the same. New models will typically be for a single activity.

column	nulls?	type	description
MODEL_ID	no	NUMBER(12,0)	Edit
SUBCLASS_VIEW	no	STRING(32)	Edit
NAME	no	STRING(255)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	Edit
SOURCE_ID		STRING(32)	Edit
VERSION_STRING		STRING(32)	Edit
VERSION_DESCRIPTION		CLOB	Edit
SECONDARY_ID		STRING(32)	Edit
BEST_PRACTICE_PARAM_GROUP_ID		TESS::ParameterGroup (NUMBER(12,0))	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Regulatory Activities
  TESS::ActivityModel
Models
  TESS::MarkovChainObs
High Volume Genome Annotation
  TESS::PredictedBindingSite
Data Transformations and Analyses
  TESS::ModelResult
  TESS::PredictionResult
Bounded Collection Grammars
  TESS::SbcgAnnotationGuide
  TESS::SbcgNonterminal
  TESS::SbcgProduction
  TESS::SbcgRecognition
  TESS::SbcgStream
Uncategorized
  TESS::SequenceFeature

Subclasses:

TESS::ModelMarkovChain
TESS::ModelSbcgGrammar

---

webmaster@gusdb.org

GUS Schema >> TESS::ModelMarkovChain (subclass of TESS::Model)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
MODEL_ID	no	NUMBER(12,0)	Edit
SUBCLASS_VIEW	no	STRING(32)	Edit
NAME	no	STRING(255)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	Edit
SOURCE_ID		STRING(32)	Edit
VERSION_STRING		STRING(32)	Edit
VERSION_DESCRIPTION		CLOB	Edit
SECONDARY_ID		STRING(32)	Edit
BEST_PRACTICE_PARAM_GROUP_ID		TESS::ParameterGroup (NUMBER(12,0))	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
IS_VARIABLE_ORDER		NUMBER(1,0)	Edit
MAXIMUM_ORDER		NUMBER(12,0)	Edit

Child tables:

Subclasses:

GUS Schema >> TESS::ModelSbcgGrammar (subclass of TESS::Model)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
MODEL_ID	no	NUMBER(12,0)	Edit
SUBCLASS_VIEW	no	STRING(32)	Edit
NAME	no	STRING(255)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	Edit
SOURCE_ID		STRING(32)	Edit
VERSION_STRING		STRING(32)	Edit
VERSION_DESCRIPTION		CLOB	Edit
SECONDARY_ID		STRING(32)	Edit
BEST_PRACTICE_PARAM_GROUP_ID		TESS::ParameterGroup (NUMBER(12,0))	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::MultinomialLabel

This table store instances of multinomial labels in a set, e.g., 'A' in DNA.

column	nulls?	type	description
MULTINOMIAL_LABEL_ID	no	NUMBER(12,0)	Primary key. Edit
MULTINOMIAL_LABEL_SET_ID	no	TESS::MultinomialLabelSet (NUMBER(12,0))	This label is part of this label set. Edit
ORDINAL	no	NUMBER	This is the ordinal of the label for placing in labels in some canonical order. Edit
LABEL	no	STRING(32)	The label. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::MultinomialLabelSet

This table stores descriptions of sets of labels, e.g., DNA or AA which are used to label multinomial observations. The actual labels are stored in `TESS.MultinomialLabel`.

column	nulls?	type	description
MULTINOMIAL_LABEL_SET_ID	no	NUMBER(12,0)	Primary key. Edit
NAME	no	STRING(20)	A suggestive name for the label set. Note that we are not encouraging verbosity. Edit
LABEL_ORDER	no	NUMBER(2,0)	Edit
SIGNATURE	no	STRING(255)	A signature of the alphabet that can be automatically generated and recognized. Edit
IS_INTEGRAL	no	NUMBER(1,0)	Will be true if labels are actually integers of some sort. Edit
CARDINALITY	no	NUMBER	The number of labels in the set. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Multinomial Models
    TESS::MultinomialLabel
    TESS::MultinomialObservationSet

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::MultinomialObservation

An observation for a `TESS.MultinomialLabel` in the context of a `TESS.MultinomialObservationSet`.

column	nulls?	type	description
MULTINOMIAL_OBS_ID	no	NUMBER(12,0)	Primary key. Edit
MULTINOMIAL_OBS_SET_ID	no	TESS::MultinomialObservationSet (NUMBER(12,0))	Observation belongs to this multinomial. Edit
MULTINOMIAL_LABEL_INDEX	no	NUMBER(12,0)	Edit
OBSERVATION		FLOAT	Observation for label. Should be counts preferably, but probabilities are allowed. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::MultinomialObservationSet

This table stores multinomial observations or probabilities.

column	nulls?	type	description
MULTINOMIAL_OBS_SET_ID	no	NUMBER	Primary key. Edit
MULTINOMIAL_LABEL_SET_ID	no	TESS::MultinomialLabelSet (NUMBER(12,0))	The labels of observations are drawn from this set. These should be shared, i.e. one for DNA, AA, etc. Edit
LABEL_ORDER	no	NUMBER	Edit
TOTAL_OBSERVATIONS	no	NUMBER	The total number of observations. Edit
IS_NORMALIZED	no	NUMBER(1,0)	Indicates whether the have observations been normalized to probabilities. No indication is stored of how this was done. Edit
ENTROPY	no	FLOAT	The information-theoretic entropy of observations. Edit
INFORMATION	no	FLOAT	The information-theoretic information content of observations. Edit
CONSENSUS		STRING(32)	A consensus string representation of observations. This should probably be considered to be suggestive. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Models
   TESS::MarkovChainObs
Multinomial Models
   TESS::MultinomialObservation

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::ArchipelagoOfInterest

Tracks a collection of regions of a genome that are of interest to a learning, analysis, or annotation project. The archipelago is a collection of islands that have been created from merged regions.

column	nulls?	type	description
ARCHIPELAGO_OF_INTEREST_ID	no	NUMBER(12,0)	Edit
NAME	no	STRING(255)	A suggestive string to identify the archipelago in output or querying. Edit
EXTERNAL_DATABASE_RELEASE_ID	no	SRes::ExternalDatabaseRelease (NUMBER(12,0))	Edit
DESCRIPTION	no	CLOB	The how and why of the archipelago's creation. Edit
ISLAND_COUNT	no	NUMBER(12,0)	How many islands are in the archipelago. Edit
REGION_COUNT	no	NUMBER(12,0)	How many regions are in the archipelago. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

High Volume Genome Annotation
   TESS::IslandOfInterest

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::IslandOfInterest

An island of regions that are of interest. Regions are merged when they are overlapping or are sufficiently close. An island is the unit of annotation, i.e., any annotation runs against an archipelago can/should process each island separately.

Note that islands apply to both strands so there is no strand information.

column	nulls?	type	description
ISLAND_OF_INTEREST_ID	no	NUMBER(12,0)	Edit
ARCHIPELAGO_OF_INTEREST_ID	no	TESS::ArchipelagoOfInterest (NUMBER(12,0))	Edit
BEGIN_POSITION	no	NUMBER(12,0)	Edit
END_POSITION	no	NUMBER(12,0)	Edit
REGION_COUNT	no	NUMBER(12,0)	How many regions are in this island. Edit
NA_SEQUENCE_ID	no	NUMBER(12,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

High Volume Genome Annotation
   TESS::RegionOfInterest

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::PredictedBindingSite

Since transcription factor binding sites are next to worthless, we store them in a separate, lighter weight, table than the `DoTS::BindingSite` and `DoTS::NALocation` that we would ordinarily use.

column	nulls?	type	description
PREDICTED_BINDING_SITE_ID	no	NUMBER(12,0)	Edit
MODEL_ID	no	TESS::Model (NUMBER(12,0))	Each site is predicted to match some model of the site. This is the link to that model. Edit
NA_SEQUENCE_ID	no	NUMBER(12,0)	This is the link to the sequence on which the prediction was made. Edit
SCORE	no	FLOAT	This is some indication of how well the site matches the model. No units are specified, but this will typically be a log-likelihood ratio or a likelihood. Edit
BEGIN_POSITION	no	NUMBER(10,0)	Edit
END_POSITION	no	NUMBER(10,0)	Edit
IS_REVERSED	no	NUMBER(1,0)	Should either be 0 or 1. When 1, the site is on the reverse strand. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::RegionOfInterest

A region is a span that surrounds a feature of interest. A link to the defining feature is maintained, as well as to the island that is created by merging nearby or overlapping regions.

Note that regions apply to both strands.

column	nulls?	type	description
REGION_OF_INTEREST_ID	no	NUMBER(12,0)	Edit
ISLAND_OF_INTEREST_ID	no	TESS::IslandOfInterest (NUMBER(12,0))	Edit
NA_Feature_ID	no	DoTS::NAFeature (NUMBER(12,0))	A link to the feature that served as an anchor for defining this region. Region boundaries and feature boundaries may different arbitrarily. Edit
BEGIN_POSITION	no	NUMBER(12,0)	Edit
END_POSITION	no	NUMBER(12,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::Footprint

This table represents a link between a sequence feature and an activity, i.e., an instance of the footprint of an activity in DNA or RNA sequence.

column	nulls?	type	description
FOOTPRINT_ID	no	NUMBER(12,0)	Primary key. Edit
ACTIVITY_ID	no	TESS::Activity (NUMBER(12,0))	A link to the `Activity` is this feature an instance of Edit
NA_Feature_ID	no	DoTS::NAFeature (NUMBER(12,0))	Link to the sequence feature Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	What database did this instance come from Edit
SOURCE_ID		STRING(32)	What is the source ID of this instance in the external database? Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Training Sets
    TESS::TrainingSetMember
Protocols
    TESS::FootprintMethodProtocol
Data Transformations and Analyses
    TESS::PredictionResult

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::TrainingSet

A selected collection of instances for an activity that can be used to train one or more models.

column	nulls?	type	description
TRAINING_SET_ID	no	NUMBER(12,0)	Primary key. Edit
ACTIVITY_ID		TESS::Activity (NUMBER(12,0))	Data relates to this activity. A null value is appropriate when the test set is a negative set that is useful for many settings. Edit
NAME	no	STRING(255)	A name for easy identification Edit
IS_POSITIVE	no	NUMBER(1,0)	is positive or negative Edit
IS_TEST	no	NUMBER(1,0)	is for testing rather than training Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Training Sets
    TESS::TrainingSetMember
Data Transformations and Analyses
    TESS::Analysis

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::TrainingSetMember

An instance in a training set

column	nulls?	type	description
TRAINING_SET_MEMBER_ID	no	NUMBER(12,0)	Primary key. Edit
TRAINING_SET_ID	no	TESS::TrainingSet (NUMBER(12,0))	Instance is a member of this training set. Edit
FOOTPRINT_ID	no	TESS::Footprint (NUMBER(12,0))	A link to the TESS.Footprint. Edit
IS_REDUNDANT		NUMBER(1,0)	Set to true when the TESS.Footprint is deemed to be redundant, i.e., covered by another TESS.Footprint. A null value indicates that its status has not been determined. A false value (0) indicates the sequence is not redundant. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> PROT::Action (subclass of PROT::Parameterizable)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
Parameterizable_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
TYPE_ID	no	NUMBER(10,0)	Edit
CHILD_PRMZ_ID		PROT::Parameterizable (NUMBER(10,0))	Edit
BIBLIOGRAPHIC_REFERENCE_ID		NUMBER(10,0)	Edit
EXTERNAL_DATABASE_RELEASE_ID		NUMBER(4,0)	Edit
SOURCE_ID		STRING(100)	Edit
NAME	no	STRING(100)	Edit
URI		STRING(100)	Edit
DESCRIPTION		STRING(4000)	Edit
ORDINAL		NUMBER(5,0)	Edit
CHILD_PROTOCOL_ID		NUMBER(10,0)	Edit
TEXT		STRING(4000)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> PROT::Parameterizable

column	nulls?	type	description
Parameterizable_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
TYPE_ID	no	NUMBER(10,0)	Edit
CHILD_PRMZ_ID		PROT::Parameterizable (NUMBER(10,0))	Edit
BIBLIOGRAPHIC_REFERENCE_ID		NUMBER(10,0)	Edit
EXTERNAL_DATABASE_RELEASE_ID		NUMBER(4,0)	Edit
SOURCE_ID		STRING(100)	Edit
NAME	no	STRING(100)	Edit
URI		STRING(100)	Edit
DESCRIPTION		STRING(4000)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Protocols
  PROT::Parameter
  PROT::Parameterizable
  PROT::ParameterizableInstance
  PROT::ParameterizableLink

Subclasses:

PROT::Software
PROT::Protocol
PROT::Hardware
PROT::Action

GUS Schema >> PROT::ActionInstance (subclass of PROT::ParameterizableInstance)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
Parameterizable_INSTANCE_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
Parameterizable_ID		PROT::Parameterizable (NUMBER(10,0))	Edit
PARENT_PI_ID		PROT::ParameterizableInstance (NUMBER(10,0))	Edit
CHILD_PI_ID		PROT::ParameterizableInstance (NUMBER(10,0))	Edit
DEVIATION		STRING(500)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> PROT::ParameterizableInstance

column	nulls?	type	description
Parameterizable_INSTANCE_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
Parameterizable_ID		PROT::Parameterizable (NUMBER(10,0))	Edit
PARENT_PI_ID		PROT::ParameterizableInstance (NUMBER(10,0))	Edit
CHILD_PI_ID		PROT::ParameterizableInstance (NUMBER(10,0))	Edit
DEVIATION		STRING(500)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Protocols
PROT::ParameterInstance
PROT::ParameterizableInstance
Platform
PROT::FeatureSet
Assay
PROT::Acquisition
PROT::Quantification

Subclasses:

PROT::ActionInstance
PROT::ProtocolInstance

GUS Schema >> TESS::FootprintMethodProtocol

How was the footprint-to-model association made?

column	nulls?	type	description
FOOTPRINT_PROTOCOL_ID	no	NUMBER(12,0)	Primary key. Edit
FOOTPRINT_ID	no	TESS::Footprint (NUMBER(12,0))	Which footprint? Edit
PROTOCOL_ID	no	NUMBER(12,0)	Which protocol? Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> PROT::Hardware (subclass of PROT::Parameterizable)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
Parameterizable_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
TYPE_ID	no	NUMBER(10,0)	Edit
CHILD_PRMZ_ID		PROT::Parameterizable (NUMBER(10,0))	Edit
BIBLIOGRAPHIC_REFERENCE_ID		NUMBER(10,0)	Edit
EXTERNAL_DATABASE_RELEASE_ID		NUMBER(4,0)	Edit
SOURCE_ID		STRING(100)	Edit
NAME	no	STRING(100)	Edit
URI		STRING(100)	Edit
DESCRIPTION		STRING(4000)	Edit
VERSION		STRING(255)	Edit
SERIAL_NUMBER		STRING(255)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::LabelMethod

This table holds additional information regarding a labeling protocol.

column	nulls?	type	description
LABEL_METHOD_ID	no	NUMBER(4,0)	Primary key. Edit
PROTOCOL_ID	no	RAD::Protocol (NUMBER(10,0))	FK to Protocol. The id of the labeling protocol this entry refers to. Edit
CHANNEL_ID	no	Study::OntologyEntry (NUMBER(10,0))	FK to Channel. The id of the channel imaging the specific label used. Edit
LABEL_USED		STRING(50)	The name of the label used. Edit
LABEL_METHOD		STRING(1000)	Free text description of any further information (other than those given in the Protocol table) about the application of that protocol with the specific dye. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Uncategorized
Study::BioMaterial

Subclasses:

webmaster@gusdb.org

GUS Schema >> PROT::Parameter

column	nulls?	type	description
Parameter_ID	no	NUMBER(10,0)	Edit
Parameterizable_ID	no	PROT::Parameterizable (NUMBER(10,0))	Edit
DATA_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
UNIT_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
NAME	no	STRING(100)	Edit
VALUE		STRING(100)	Edit
VALUE_CLASS_OE_ID	no	Study::OntologyEntry (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Protocols
PROT::ParameterInstance

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> PROT::ParameterInstance

column	nulls?	type	description
Parameter_INSTANCE_ID	no	NUMBER(12,0)	Edit
Parameterizable_INSTANCE_ID		PROT::ParameterizableInstance (NUMBER(12,0))	Edit
Parameter_ID	no	PROT::Parameter (NUMBER(12,0))	Edit
VALUE	no	STRING(100)	Edit
VALUE_ONTOLOGY_ENTRY_ID	no	Study::OntologyEntry (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> PROT::ParameterizableLink

column	nulls?	type	description
Parameterizable_LINK_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
PARENT_PRMZ_ID	no	PROT::Parameterizable (NUMBER(10,0))	Edit
CHILD_PRMZ_ID	no	PROT::Parameterizable (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

PROT::ProtocolAction
PROT::ProtocolHardware
PROT::ProtocolSoftware

---

webmaster@gusdb.org

GUS Schema >> PROT::Protocol (subclass of PROT::Parameterizable)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
Parameterizable_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
TYPE_ID	no	NUMBER(10,0)	Edit
CHILD_PRMZ_ID		PROT::Parameterizable (NUMBER(10,0))	Edit
BIBLIOGRAPHIC_REFERENCE_ID		NUMBER(10,0)	Edit
EXTERNAL_DATABASE_RELEASE_ID		NUMBER(4,0)	Edit
SOURCE_ID		STRING(100)	Edit
NAME	no	STRING(100)	Edit
URI		STRING(100)	Edit
DESCRIPTION		STRING(4000)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> PROT::ProtocolAction (subclass of PROT::ParameterizableLink)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
Parameterizable_LINK_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
PARENT_PRMZ_ID	no	PROT::Parameterizable (NUMBER(10,0))	Edit
CHILD_PRMZ_ID	no	PROT::Parameterizable (NUMBER(10,0))	Edit
PROTOCOL_ID	no	NUMBER(10,0)	Edit
Action_ID	no	NUMBER(10,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> PROT::ProtocolHardware (subclass of PROT::ParameterizableLink)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
Parameterizable_LINK_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
PARENT_PRMZ_ID	no	PROT::Parameterizable (NUMBER(10,0))	Edit
CHILD_PRMZ_ID	no	PROT::Parameterizable (NUMBER(10,0))	Edit
PROTOCOL_ID	no	NUMBER(10,0)	Edit
Hardware_ID	no	NUMBER(10,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> PROT::ProtocolInstance (subclass of PROT::ParameterizableInstance)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
Parameterizable_INSTANCE_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
Parameterizable_ID		PROT::Parameterizable (NUMBER(10,0))	Edit
PARENT_PI_ID		PROT::ParameterizableInstance (NUMBER(10,0))	Edit
CHILD_PI_ID		PROT::ParameterizableInstance (NUMBER(10,0))	Edit
DEVIATION		STRING(500)	Edit

Child tables:

Subclasses:

GUS Schema >> RAD::ProtocolParam

Holds information to describe the parameters used in the protocol

column	nulls?	type	description
PROTOCOL_PARAM_ID	no	NUMBER(10,0)	primary key Edit
PROTOCOL_ID	no	RAD::Protocol (NUMBER(10,0))	FK to Protocol Edit
NAME	no	STRING(100)	"A unique name for the parameter, with respect to the owning Protocol." Edit
DATA_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	"The data type of the parameter value (e.g. ""volume"")" Edit
UNIT_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	The unit for the value Edit
VALUE		STRING(100)	A default numerical or string value for parameter Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Data Transformations and Analyses
    RAD::AnalysisParam
Assay
    RAD::AcquisitionParam
    RAD::AssayParam
    RAD::QuantificationParam
Biomaterials
    RAD::TreatmentParam

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::ProtocolQCParam

Holds information about quality control parameters for Protocols

column	nulls?	type	description
PROTOCOL_QC_PARAM_ID	no	NUMBER(5,0)	Primary key Edit
PROTOCOL_ID	no	RAD::Protocol (NUMBER(10,0))	Foreign key to RAD3.Protocol Edit
NAME	no	STRING(100)	"A unique name for the parameter, with respect to the owning Protocol." Edit
DATA_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	"The data type of the parameter value (e.g. ""volume"")" Edit
UNIT_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	The unit for the value Edit
VALUE		STRING(100)	A default numerical or string value for parameter Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Data Transformations and Analyses
   RAD::AnalysisQCParam

Subclasses:

webmaster@gusdb.org

gusdb.org | Schema Browser

[GUS Schema](#) >> **PROT::ProtocolSoftware** (subclass of [PROT::ParameterizableLink](#))

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
Parameterizable_LINK_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
PARENT_PRMZ_ID	no	[PROT::Parameterizable](#) (NUMBER(10,0))	Edit
CHILD_PRMZ_ID	no	[PROT::Parameterizable](#) (NUMBER(10,0))	Edit
PROTOCOL_ID	no	NUMBER(10,0)	Edit
Software_ID	no	NUMBER(10,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::ProtocolStep

This table is used to link a protocol that consists of an ordered series of component protocols to its components.

column	nulls?	type	description
PROTOCOL_STEP_ID	no	NUMBER(10,0)	primary key Edit
PARENT_PROTOCOL_ID	no	RAD::Protocol (NUMBER(10,0))	FK pointing to the protocol representing the series. Edit
CHILD_PROTOCOL_ID	no	RAD::Protocol (NUMBER(10,0))	FK pointing to the protocol representing a component of the series. Edit
ORDER_NUM	no	NUMBER(3,0)	The order of this component in this series, e.g. 1 if it's the first protocol in the series, 2 if it's the second, etc. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> PROT::Software (subclass of PROT::Parameterizable)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
Parameterizable_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
TYPE_ID	no	NUMBER(10,0)	Edit
CHILD_PRMZ_ID		PROT::Parameterizable (NUMBER(10,0))	Edit
BIBLIOGRAPHIC_REFERENCE_ID		NUMBER(10,0)	Edit
EXTERNAL_DATABASE_RELEASE_ID		NUMBER(4,0)	Edit
SOURCE_ID		STRING(100)	Edit
NAME	no	STRING(100)	Edit
URI		STRING(100)	Edit
DESCRIPTION		STRING(4000)	Edit
VERSION		STRING(255)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::Analysis

The `Analysis` records that fact that either a learning or prediction run was made.

column	nulls?	type	description
ANALYSIS_ID	no	NUMBER(12,0)	Primary key. Edit
POSITIVE_TRAINING_SET_ID	no	TESS::TrainingSet (NUMBER(12,0))	A link to the positive `TrainingSet` that the analysis was run on. Edit
NEGATIVE_TRAINING_SET_ID		TESS::TrainingSet (NUMBER(12,0))	An optional link to the negative `TrainingSet` that the analysis was run on. Edit
PROTOCOL_ID	no	NUMBER(12,0)	A link to a cv/ontology of protocols. Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	The review status of the results of this analysis. Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	The database this analysis was drawn from. Will be null only if generated internally. Edit
MODEL_COUNT	no	NUMBER(6,0)	The number of models learned. Edit
PREDICTION_COUNT	no	NUMBER(6,0)	The number of predictions made. Edit
Parameter_GROUP_ID		TESS::ParameterGroup (NUMBER(12,0))	A (super)group of parameters used or learned in the analysis. If multiple grammars are used, then this could point to a group above the individual groups that the grammars point to. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Data Transformations and Analyses
TESS::ModelResult
TESS::PredictionResult

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::AnalysisInput

Each entry in this table represents an input of a specific down-stream gene expression analysis. Each input to an analysis is a logical group.

column	nulls?	type	description
ANALYSIS_INPUT_ID	no	NUMBER(10,0)	primary_key Edit
LOGICAL_GROUP_ID	no	RAD::LogicalGroup (NUMBER(8,0))	FK to LogicalGroup. The input this entry refers to. Edit
ANALYSIS_ID	no	RAD::Analysis (NUMBER(8,0))	FK to Analysis. The analysis this entry refers to. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::AnalysisParam

This table is used to enter values of down-stream analysis protocol parameters for a specific down-stream gene expression analysis using a given protocol. Precisely this table can be used to either over-write the default value of a parameter (as stored in the ProtocolParam table) or to enter a value of a parameter for which a default had not been set.

column	nulls?	type	description
ANALYSIS_PARAM_ID	no	NUMBER(10,0)	primary_key Edit
ANALYSIS_ID	no	RAD::Analysis (NUMBER(8,0))	FK to Analysis. The specific analysis to which this parameter value refers. Edit
PROTOCOL_PARAM_ID	no	RAD::ProtocolParam (NUMBER(10,0))	FK to ProtocolParam. The protocol parameter whose value is entered (or whose default is overwritten). Edit
VALUE	no	STRING(100)	The value of the parameter in this particular analysis. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

gusdb.org | Schema Browser

GUS Schema >> RAD::AnalysisQCParam

This table is used to enter values of down-stream analysis protocol quality control parameters for a specific down-stream gene expression analysis using a given protocol. Precisely this table can be used to either over-write the default value of a parameter (as stored in the ProtocolQCParam table) or to enter a value of a parameter for which a default had not been set.

column	nulls?	type	description
ANALYSIS_QC_PARAM_ID	no	NUMBER(10,0)	primary_key Edit
ANALYSIS_ID	no	RAD::Analysis (NUMBER(8,0))	FK to Analysis. The specific analysis to which this parameter value refers. Edit
PROTOCOL_QC_PARAM_ID	no	RAD::ProtocolQCParam (NUMBER(10,0))	FK to ProtocolQCParam. The protocol quality control parameter whose value is entered (or whose default is overwritten). Edit
VALUE	no	STRING(100)	The value of the parameter in this particular analysis. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::AnalysisResult

Holds the results of a down-stream gene expression analysis.

column	nulls?	type	description
ANALYSIS_RESULT_ID	no	NUMBER(12,0)	primary key Edit
SUBCLASS_VIEW	no	STRING(27)	name of the view Edit
ANALYSIS_ID	no	RAD::Analysis (NUMBER(8,0))	FK to Analysis. The analysis this result refers to. Edit
TABLE_ID		Core::TableInfo (NUMBER(5,0))	Edit
ROW_ID		NUMBER(12,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

RAD::SAM
RAD::SpliceArrayAnalysisResult
RAD::ArrayStatTwoConditions
RAD::DataTransformationResult
RAD::HQSpecificity

webmaster@gusdb.org

GUS Schema >> RAD::ArrayStatTwoConditions (subclass of RAD::AnalysisResult)

This view of AnalysisResultImp is for the results of analyses done with the ArrayStat software by Imaging Research Inc. to compare two conditions in order to detect differentially expressed genes.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ANALYSIS_RESULT_ID	no	NUMBER(12,0)	primary key Edit
SUBCLASS_VIEW	no	STRING(27)	name of the view Edit
ANALYSIS_ID	no	RAD::Analysis (NUMBER(8,0))	FK to Analysis. The analysis this result refers to. Edit
TABLE_ID		Core::TableInfo (NUMBER(5,0))	Edit
ROW_ID		NUMBER(12,0)	Edit
MEAN_1		FLOAT	The mean of the processed (after the second normalization step embedded in this analysis) replicate values in group 1 for the (composite) element identified by row_id. Edit
SD_1		FLOAT	The standard deviation of the processed (after the second normalization step embedded in this analysis) replicate values in group 1 for the (composite) element identified by row_id. Edit
N_1		NUMBER(8,0)	The number of non-outlier data points for group 1 for the (composite) element identified by row_id. Edit
MEAN_2		FLOAT	The mean of the processed (after the second normalization step embedded in this analysis) replicate values in group 2 for the (composite) element identified by row_id. Edit
SD_2		FLOAT	The standard deviation of the processed (after the second normalization step embedded in this analysis) replicate values in group 2 for the (composite) element identified by row_id. Edit
N_2		NUMBER(8,0)	The number of non-outlier data points for group 2 for the (composite) element identified by row_id. Edit
P_VALUE		FLOAT	The (un-adjusted) 2-tailed p-value of the statistic used in this application of the software, for the (composite) element identified by row_id. Edit
SIGNIFICANCE		STRING(10)	One of "yes" or "no": "yes" indicates that the (composite) element identified by row_id shows significant differential expression between group 1 and group 2 after correction for multiple testing done with the method and alpha value specified in the analysis parameters for this application. Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::AssayAnalysis

This table is used to link an analysis to all assays that were involved in it.

column	nulls?	type	description
ASSAY_ANALYSIS_ID	no	NUMBER(12,0)	primary key Edit
ASSAY_ID	no	RAD::Assay (NUMBER(8,0))	FK pointing to one of the relevant assays. Edit
ANALYSIS_ID	no	RAD::Analysis (NUMBER(8,0))	FK pointing to the analysis. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> PROT::DataTransformation

column	nulls?	type	description
DATA_TRANSFORMATION_IO_ID	no	NUMBER(8,0)	Edit
QUANTIFICATION_ID		PROT::Quantification (NUMBER(8,0))	Edit
INPUT_QUANTIFICATION_ID		PROT::Quantification (NUMBER(8,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::DataTransformationResult (subclass of RAD::AnalysisResult)

This view of RAD3::AnalysisResultImp is used to store the results of array data pre-processing (e.g. normalization and other series of up-stream transformations).

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ANALYSIS_RESULT_ID	no	NUMBER(12,0)	primary key Edit
SUBCLASS_VIEW	no	STRING(27)	name of the view Edit
ANALYSIS_ID	no	RAD::Analysis (NUMBER(8,0))	FK to Analysis. The analysis this result refers to. Edit
TABLE_ID		Core::TableInfo (NUMBER(5,0))	Edit
ROW_ID		NUMBER(12,0)	Edit
FLOAT_VALUE		FLOAT	If the transformation result is of float type, it should be stored here. Edit
INT_VALUE		NUMBER(8,0)	If the transformation result is of integer type, it should be stored here. Edit
STRING_VALUE		STRING(200)	If the transformation result is of string type, it should be stored here. Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::HQSpecificity (subclass of RAD::AnalysisResult)

This view of AnalysisResultImp is for the results of analyses done with the H–Q specificity protocol. The output of such an analysis consists of a tree (e.g. based on an anatomical hierarchy) for each (composite) element examined, where the nodes of the tree store the values for the global and local Shannon entropy and weights.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ANALYSIS_RESULT_ID	no	NUMBER(12,0)	primary key Edit
SUBCLASS_VIEW	no	STRING(27)	name of the view Edit
ANALYSIS_ID	no	RAD::Analysis (NUMBER(8,0))	FK to Analysis. The analysis this result refers to. Edit
TABLE_ID		Core::TableInfo (NUMBER(5,0))	Edit
ROW_ID		NUMBER(12,0)	Edit
NODE_NUMBER		NUMBER(8,0)	The number corresponding to the node in the tree to which these results refer. Edit
PARENT_NODE_NUMBER		NUMBER(8,0)	The number corresponding to the node in the tree which is the parent of the current node. Edit
LEAF_COUNT		NUMBER(8,0)	The number of leaf nodes at or below this node. Is one for every leaf node and strictly greater than one otherwise (assuming the tree has no 'only' children). Edit
H_G		FLOAT	The value of the global Shannon entropy at the current node, i.e. the entropy w.r.t. the leaves in the tree. Edit
H_L		FLOAT	The value of the local Shannon entropy at the current node, i.e. the entropy w.r.t. the immediate children of this node. Edit
I_G		FLOAT	The value of the global Shannon information at the current node, i.e., the information content w.r.t. the leaves of the tree. This is a derived value computed from h_g and node_number and stored for efficiency. Edit
W		FLOAT	The weight of the current node. Edit
MINUS_LOG2_W		FLOAT	The negative log (base 2) of w. Derived quantity stored for query efficiency. Edit
EXPRESSION_LEVEL		FLOAT	Edit
MAX_EXPRESSION_LEVEL_BELOW		FLOAT	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> RAD::LogicalGroup

This table holds information about a group of any kind of entities in RAD tables. For example, a set of process invocations which have been used with a clutering algorithm, typically used as the input for the AnalysisInput table.

column	nulls?	type	description
LOGICAL_GROUP_ID	no	NUMBER(8,0)	Primary key Edit
NAME	no	STRING(100)	The name identifying this group. Edit
CATEGORY		STRING(50)	the name identifying the entities in this group, typically the name of the table in which these entities exist. for example, assay, spot, quantification, etc. Edit
DESCRIPTION		STRING(1000)	A brief description for this group. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Data Transformations and Analyses
    RAD::AnalysisInput
    RAD::LogicalGroupLink

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> RAD::LogicalGroupLink

This table is used to link a group member to its group defined in LogicalGroup.

column	nulls?	type	description
LOGICAL_GROUP_LINK_ID	no	NUMBER(10,0)	Primary key. Edit
LOGICAL_GROUP_ID	no	RAD::LogicalGroup (NUMBER(8,0))	FK to LogicalGroup. The group to which this entry refers. Edit
TABLE_ID	no	Core::TableInfo (NUMBER(5,0))	FK to Core.TableInfo. The table where the group member exists, which this entry is linking to the group. Edit
ROW_ID	no	NUMBER(12,0)	The primary key in the table pointed by the table_id, which identifies this group memember in that table. Edit
ORDER_NUM		NUMBER(8,0)	indicates the position of the member within the list if the list is ordered Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

gusdb.org | Schema Browser

GUS Schema >> TESS::ModelResult

column	nulls?	type	description
MODEL_RESULT_ID	no	NUMBER(12,0)	Primary key. Edit
MODEL_ID	no	TESS::Model (NUMBER(12,0))	Edit
ANALYSIS_ID	no	TESS::Analysis (NUMBER(12,0))	Edit
Parameter_GROUP_ID		TESS::ParameterGroup (NUMBER(12,0))	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	The review status of the model results of this analysis. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::PredictionResult

column	nulls?	type	description
PREDICTION_RESULT_ID	no	NUMBER(12,0)	Primary key. Edit
MODEL_ID	no	TESS::Model (NUMBER(12,0))	Edit
FOOTPRINT_ID	no	TESS::Footprint (NUMBER(12,0))	A link to the Footprint associated with the model. Edit
ANALYSIS_ID	no	TESS::Analysis (NUMBER(12,0))	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	The review status of the prediction results of this analysis. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::SAM (subclass of RAD::AnalysisResult)

This view of AnalysisResultImp is for the results of analyses done with SAM (Statistical Analysis of Microarrays). The output of a SAM analysis consists of a list of significant (composite) elements with attached d statistics and q-values.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ANALYSIS_RESULT_ID	no	NUMBER(12,0)	primary key Edit
SUBCLASS_VIEW	no	STRING(27)	name of the view Edit
ANALYSIS_ID	no	RAD::Analysis (NUMBER(8,0))	FK to Analysis. The analysis this result refers to. Edit
TABLE_ID		Core::TableInfo (NUMBER(5,0))	Edit
ROW_ID		NUMBER(12,0)	Edit
D		FLOAT	The value of the SAM d statistic for this (composite) element. Edit
R		FLOAT	The numerator of d. Edit
S_PLUS_S0		FLOAT	The denominator of d. Edit
Q_VALUE_PERC		FLOAT	The lowest FDR at which the gene is called significant. This is expressed as a percentage. Edit
FOLD_CHANGE		FLOAT	The ratio of the average expression levels of a gene under condition 1 and condition 2. These averages refer to the raw (unlogged) data. Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::SpliceArrayAnalysisResult (subclass of RAD::AnalysisResult)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ANALYSIS_RESULT_ID	no	NUMBER(12,0)	primary key Edit
SUBCLASS_VIEW	no	STRING(27)	name of the view Edit
ANALYSIS_ID	no	RAD::Analysis (NUMBER(8,0))	FK to Analysis. The analysis this result refers to. Edit
TABLE_ID		Core::TableInfo (NUMBER(5,0))	Edit
ROW_ID		NUMBER(12,0)	Edit
CHANNEL1		FLOAT	Edit
CHANNEL2		FLOAT	Edit
MEAN_LOG_RATIO		FLOAT	Edit
RATIO_ERROR		FLOAT	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::ParameterGroup

A group of parameters and subgroups.

column	nulls?	type	description
Parameter_GROUP_ID	no	NUMBER(12,0)	Primary key. Edit
NAME	no	STRING(255)	A suggestive name for the group. Used in combination with `TESS.ParameterSubgroup` entries these can create a path to parameter value. Edit
DESCRIPTION		CLOB	A fuller description of the group. Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Extent of reviewing Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Models
    TESS::Model
Data Transformations and Analyses
    TESS::Analysis
    TESS::ModelResult
Learning Models
    TESS::ParameterSubgroup
    TESS::ParameterValue

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::ParameterSubgroup

This is a means of combining `TESS.ParameterGroup` rows to reuse and override certain parameters.

column	nulls?	type	description
Parameter_SUBGROUP_ID	no	NUMBER(12,0)	Primary key. Edit
PARENT_GROUP_ID	no	TESS::ParameterGroup (NUMBER(12,0))	Link to parent group. Edit
CHILD_GROUP_ID	no	TESS::ParameterGroup (NUMBER(12,0))	Link to child group. Edit
ORDINAL	no	NUMBER	A means of putting groups in order. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::ParameterValue

A parameter with name, type and value.

column	nulls?	type	description
Parameter_VALUE_ID	no	NUMBER(12,0)	Primary key. Edit
Parameter_GROUP_ID	no	TESS::ParameterGroup (NUMBER(12,0))	The parameter belongs to this set of test conditions. Edit
ORDINAL	no	NUMBER	A means of putting parameters in order. Could be used to implement arrays but that may not be recommended. Edit
NAME	no	STRING(255)	Edit
TYPE_ID		Core::AlgorithmParamKeyType (NUMBER(12,0))	The type of the parameter, is untyped unless a reference is made. Edit
VALUE		STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

[GUS Schema](#) >> TESS::ParserItem

ParserItem is the link between Recognitin and DoTS.NaFeature rows. Some of the parse will be mirrored in DoTS using various NAFeature views and NAFeatureRelationship but not all of it. How much gets mirrored is under control of the material stored in the SbcgAnnotationGuide table. The ParserItem* family of table stores the complete parse. These tables form a parse graph with labeled edges.

column	nulls?	type	description
PARSER_ITEM_ID	no	NUMBER(12,0)	Primary key. Edit
SBCG_RECOGNITION_ID	no	TESS::SbcgRecognition (NUMBER(12,0))	A link to the SbcgRecognition entity that matched. Edit
NA_Feature_ID		DoTS::NAFeature (NUMBER(12,0))	An optional link to some kind of NAFeature view. Is null when no annotation has been made or found. Edit
AA_Feature_ID		NUMBER(12,0)	An optional link to some kind of AAFeature view. Is null when no annotation has been made or found. Looking forward to parsing AA sequences. This may be cheesy. Suggestions. Anyone. Edit
INSTANCE_COUNT	no	NUMBER(6,0)	The number of times this recognition has appeared in the expansion. This is really only useful for bag productions. The first appearance is counted as one. Edit
BEGIN_COORD	no	NUMBER(12,0)	Begining of item on sequence. This attribute follows standard GUS conventions. Edit
END_COORD	no	NUMBER(12,0)	End of item on sequence. This attribute follows standard GUS conventions. Edit
IS_REVERSED	no	NUMBER(1,0)	Orientation of item on sequence. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Learning Models
    [TESS::ParserItemLink](#)

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> TESS::ParserItemEdgeType

This is a CV for the different types of relations between recognition elements, i.e., part-of, contained, join, meet.

column	nulls?	type	description
PARSER_ITEM_EDGE_TYPE_ID	no	NUMBER(12,0)	Primary key. Edit
NAME	no	STRING(16)	The name of the operation. Edit
DESCRIPTION	no	CLOB	A brief explanation of the operation Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Learning Models
   TESS::ParserItemLink

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::ParserItemLink

This table links `ParserItem` row together to form the graph. The edges are labeled or typed so there is a link to the type table.

column	nulls?	type	description
PARSER_ITEM_LINK_ID	no	NUMBER(12,0)	Edit
PARENT_PARSER_ITEM_ID	no	TESS::ParserItem (NUMBER(12,0))	Primary key. Edit
CHILD_PARSER_ITEM_ID	no	TESS::ParserItem (NUMBER(12,0))	Edit
PARSER_ITEM_EDGE_TYPE_ID	no	TESS::ParserItemEdgeType (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::SbcgAnnotationFilter

This table contains the information needed to select a path term, either in the context of a path bound or as a RHS term.

column	nulls?	type	description
SBCG_ANNOTATION_FILTER_ID	no	NUMBER(12,0)	Primary key. Edit
SBCG_RECOG_PATH_EXPRESSION_ID	no	TESS::SbcgRecognition (NUMBER(12,0))	Point to the path this filter belongs to. Edit
ORDINAL	no	NUMBER(6,0)	The position of this term in the path counted from 1. Edit
SBCG_PATH_TERM_REL_TYPE_ID		TESS::SbcgPathTermRelationType (NUMBER(12,0))	Operation used to relate this term to previous term. Is null then the filter belongs to a Recognition. Edit
SBCG_STREAM_NAME		STRING(32)	Point to the stream we are searching Edit
NUMBER_OF_FILTER_TERMS	no	NUMBER(3,0)	Record the number of terms. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Bounded Collection Grammars
    TESS::SbcgAnnotationFilterTerm

Subclasses:

---

webmaster@gusdb.org

[GUS Schema](#) >> **TESS::SbcgAnnotationFilterTerm**

This table contains comparisons between annotation attributes and constant values. These are linked to a `TESS.SbcgRecogAnnotationFilter` to make a complete filter.

column	nulls?	type	description
SCBG_ANNOTATION_FILTER_TERM_ID	no	NUMBER(12,0)	Primary key. Edit
SBCG_ANNOTATION_FILTER_ID	no	[TESS::SbcgAnnotationFilter](#) (NUMBER(12,0))	A link to a `TESS::SbcgAnnotationFilter` that will use this filter term. Edit
ORDINAL	no	NUMBER(4,0)	Position in list of comparisions. Edit
ATTRIBUTE_NAME	no	STRING(32)	Comparison should be made to this attribute of the annotation. Edit
SBCG_COMPARISON_TYPE_ID	no	[TESS::SbcgComparisonType](#) (NUMBER(12,0))	Type of comparison to perform. Edit
IS_NUMERIC_VALUE	no	NUMBER(1,0)	True when the value is numeric. Edit
CONSTANT_VALUE	no	STRING(255)	Compare to this constant value. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> TESS::SbcgAnnotationGuide

This table contains information about creating annotation objects from elements of a grammar upon a successful parse.

column	nulls?	type	description
SBCG_ANNOTATION_GUIDE_ID	no	NUMBER(12,0)	Primary key. Edit
MODEL_ID	no	TESS::Model (NUMBER(12,0))	Edit
SOURCE_CODE	no	CLOB	This field contains the source code for the stream definition. Edit
TYPE_VALUE		STRING(255)	The type name of the annotation. Edit
CATEGORY_VALUE		STRING(255)	The category of the annotation. Edit
NAME_VALUE		STRING(255)	The name of the annotation. Edit
ID_VALUE		STRING(255)	The id of the annotation. Edit
METHOD_VALUE		STRING(255)	The method of the annotation. Edit
MODELID_VALUE		STRING(255)	The modelid of the annotation. Edit
MODEL_VALUE		STRING(255)	The model of the annotation. Edit
SCORE_VALUE		STRING(255)	The score of the annotation. Edit
REVIEWED_VALUE		STRING(255)	The reviewed of the annotation. Edit
START_VALUE		STRING(255)	The start of the annotation. Edit
END_VALUE		STRING(255)	The end of the annotation. Edit
SENSE_VALUE		STRING(255)	The sense of the annotation. Edit
LENGTH_VALUE		STRING(255)	The length of the annotation. Edit
INDEX_VALUE		STRING(255)	The index of the annotation. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Bounded Collection Grammars
 TESS::SbcgProduction
 TESS::SbcgRhsTerm

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::SbcgComparisonType

This table contains comparision operators to used when comparing annotation attribute values to constants.

column	nulls?	type	description
SBCG_COMPARISON_TYPE_ID	no	NUMBER(12,0)	Primary key. Edit
NAME	no	STRING(32)	The relation name. Edit
DESCRIPTION		STRING(255)	A brief description. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Bounded Collection Grammars
  TESS::SbcgAnnotationFilterTerm

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> TESS::SbcgNonterminal

This table stores all the nonterminals for a grammar.

column	nulls?	type	description
SBCG_NONTERMINAL_ID	no	NUMBER(12,0)	Primary key. Edit
SBGC_GRAMMAR_ID	no	TESS::Model (NUMBER(12,0))	The grammar containing this production. Edit
NAME	no	STRING(255)	The name of the nonterminal. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Bounded Collection Grammars
TESS::SbcgProduction
TESS::SbcgRecognition

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::SbcgPathTermRelationType

This table contains controlled vocabulary terms for path bounding term relations.

column	nulls?	type	description
SBCG_PATH_TERM_REL_TYPE_ID	no	NUMBER(12,0)	Primary key. Edit
NAME	no	STRING(32)	The relation name. Edit
DESCRIPTION		STRING(255)	A brief description. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Bounded Collection Grammars
   TESS::SbcgAnnotationFilter

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> TESS::SbcgProduction

This table stores productions in a grammar. The complete source code for the Details of the production are stored in related tables SbcgRhsTerm, SbcgAnnotationGuide, SbcgRecognition, and SbcgNonterminal. A production is defined as a single expansion for a nonterminal. Thus A -> B C | D E; contains two productions.

column	nulls?	type	description
SBCG_PRODUCTION_ID	no	NUMBER(12,0)	Primary key. Edit
SBGC_GRAMMAR_ID	no	TESS::Model (NUMBER(12,0))	The grammar containing this production. Edit
ORDINAL	no	NUMBER(12,0)	This field is used to order the productions. Edit
SOURCE_CODE	no	CLOB	This field contains the source code for the production definition. Edit
COMMENT_STRING		CLOB	This is a comment attached to the production. Edit
SBCG_NONTERMINAL_ID	no	TESS::SbcgNonterminal (NUMBER(12,0))	This is a link to the nonterminal the production expands. Edit
LG_PROBABILITY	no	FLOAT	This is the log (base 2)probability of this expansion amoung all productions with the same LHS. The sum of 2^p of these probabilities (for each nonterminal) should be one in a well formed grammar. They should all be nonpositive. Edit
SBCG_PRODUCTION_TYPE_ID	no	TESS::SbcgProductionType (NUMBER(12,0))	This is the type (normal, list, set or bag) of the production. Edit
RHS_TERMS_N	no	NUMBER(10,0)	This is the number of rhs terms in the production. This must be kept in agreement with the actual number of SbcgRhsTerm rows referencing this row. This value should be zero if an only if the production is a null production. Edit
SBCG_ANNOTATION_GUIDE_ID		TESS::SbcgAnnotationGuide (NUMBER(12,0))	This is a link to an annotation guide for the production. Edit
SIZE_BOUND_N		NUMBER(12,0)	This is the maximum size of the legal expansion for the production. If the value is null, then the production has no size bound. Note that path bound information is stored in SbcgPathBound. Edit
SBCG_PATH_BOUND_ID		TESS::SbcgRecognition (NUMBER(12,0))	This is a path expression that bounds the expansion of the RHS. Edit
MINIMAL	no	NUMBER(1,0)	This indicates if matches to the production are required to be minimal or not. Minimal matches do not contain any shorter interval which will also match. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Bounded Collection Grammars
    TESS::SbcgRhsTerm

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::SbcgProductionType

This table contains a list of controlled vocabulary terms for the different types of productions in a SBCG.

column	nulls?	type	description
SBCG_PRODUCTION_TYPE_ID	no	NUMBER(12,0)	Primary key. Edit
TYPE_NAME	no	STRING(16)	Name of the type of production. Initial values are normal, list, set, and bag. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Bounded Collection Grammars
TESS::SbcgProduction

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> TESS::SbcgRecogConsensusString (subclass of TESS::SbcgRecognition)

This table stores consensus strings as recognition primitives in a SBCG.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
SBCG_RECOGNITION_ID	no	NUMBER(12,0)	Primary key. Edit
SUBCLASS_VIEW	no	STRING(32)	Edit
MODEL_ID	no	TESS::Model (NUMBER(12,0))	Edit
SBCG_NONTERMINAL_ID		TESS::SbcgNonterminal (NUMBER(12,0))	Edit
SBCG_STREAM_ID		TESS::SbcgStream (NUMBER(12,0))	Edit
PARENT_SBCG_RECOGNITION_ID		TESS::SbcgRecognition (NUMBER(12,0))	Edit
VALUE		STRING(255)	Consensus string to match. Edit
MISMATCHES_ALLOWED		NUMBER	Number of allowable mismatches. This is not yet supported by the parser or `Reader` format. Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::SbcgRecognition

Superclass for all `SbcgRecognition` views.

column	nulls?	type	description
SBCG_RECOGNITION_ID	no	NUMBER(12,0)	Primary key. Edit
SUBCLASS_VIEW	no	STRING(32)	Edit
MODEL_ID	no	TESS::Model (NUMBER(12,0))	Edit
SBCG_NONTERMINAL_ID		TESS::SbcgNonterminal (NUMBER(12,0))	Edit
SBCG_STREAM_ID		TESS::SbcgStream (NUMBER(12,0))	Edit
PARENT_SBCG_RECOGNITION_ID		TESS::SbcgRecognition (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Learning Models
  TESS::ParserItem
Bounded Collection Grammars
  TESS::SbcgAnnotationFilter
  TESS::SbcgProduction
  TESS::SbcgRecogMultiObs
  TESS::SbcgRecognition
  TESS::SbcgRhsTerm

Subclasses:

TESS::SbcgRecogPosition
TESS::SbcgRecogGap
TESS::SbcgRecogWeightMatrix
TESS::SbcgRecogConsensusString
TESS::SbcgRecogString
TESS::SbcgRecogPathExpression

[GUS Schema](#) >> **TESS::SbcgRecogGap** (subclass of [TESS::SbcgRecognition](#))

This view represents a gap. The size (range) is set by the count information on the *TESS.SbcgRhsTerm*.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
SBCG_RECOGNITION_ID	no	NUMBER(12,0)	Primary key. [Edit](#)
SUBCLASS_VIEW	no	STRING(32)	[Edit](#)
MODEL_ID	no	[TESS::Model](#) (NUMBER(12,0))	[Edit](#)
SBCG_NONTERMINAL_ID		[TESS::SbcgNonterminal](#) (NUMBER(12,0))	[Edit](#)
SBCG_STREAM_ID		[TESS::SbcgStream](#) (NUMBER(12,0))	[Edit](#)
PARENT_SBCG_RECOGNITION_ID		[TESS::SbcgRecognition](#) (NUMBER(12,0))	[Edit](#)

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> TESS::SbcgRecogMultiObs

column	nulls?	type	description
SBCG_RECOG_MULTI_OBS_ID	no	NUMBER(12,0)	Edit
SBCG_RECOGNITION_ID	no	TESS::SbcgRecognition (NUMBER(12,0))	A link to the SbcgRecognition the observation belongs to. This is expected to be a weight matrix. Edit
MULTINOMIAL_OBSERVATION_SET_ID	no	NUMBER(12,0)	A link to the observation Edit
ORDINAL	no	NUMBER	The position of the observation in the recognition. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::SbcgRecogPathExpression (subclass of TESS::SbcgRecognition)

Rows in this table form the nub of a path expression that can either be used as a bound or as a RHS term.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
SBCG_RECOGNITION_ID	no	NUMBER(12,0)	Primary key. Edit
SUBCLASS_VIEW	no	STRING(32)	Edit
MODEL_ID	no	TESS::Model (NUMBER(12,0))	Edit
SBCG_NONTERMINAL_ID		TESS::SbcgNonterminal (NUMBER(12,0))	Edit
SBCG_STREAM_ID		TESS::SbcgStream (NUMBER(12,0))	Edit
PARENT_SBCG_RECOGNITION_ID		TESS::SbcgRecognition (NUMBER(12,0))	Edit
LENGTH		NUMBER	The number of terms in the path. Edit
ANCHOR_POINT_BEGIN		NUMBER(1,0)	Use beginning of final interval(s). Edit
ANCHOR_POINT_CENTER		NUMBER(1,0)	Use center of final interval(s). Edit
ANCHOR_POINT_END		NUMBER(1,0)	Use end of final interval(s). Edit

Child tables:

Subclasses:

GUS Schema >> TESS::SbcgRecogPosition (subclass of TESS::SbcgRecognition)

This view represents absolute or relative positions as patterns to match.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
SBCG_RECOGNITION_ID	no	NUMBER(12,0)	Primary key. Edit
SUBCLASS_VIEW	no	STRING(32)	Edit
MODEL_ID	no	TESS::Model (NUMBER(12,0))	Edit
SBCG_NONTERMINAL_ID		TESS::SbcgNonterminal (NUMBER(12,0))	Edit
SBCG_STREAM_ID		TESS::SbcgStream (NUMBER(12,0))	Edit
PARENT_SBCG_RECOGNITION_ID		TESS::SbcgRecognition (NUMBER(12,0))	Edit
IS_RELATIVE		NUMBER(1,0)	Is this a position relative to an enclosing path bound interval? Edit
VALUE		NUMBER	Numeric value of position. 1–based and negative numbers count from end of string. Zero is not a legal value. Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> TESS::SbcgRecogString (subclass of TESS::SbcgRecognition)

This table stores strings as recognition primitives in a SBCG.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
SBCG_RECOGNITION_ID	no	NUMBER(12,0)	Primary key. Edit
SUBCLASS_VIEW	no	STRING(32)	Edit
MODEL_ID	no	TESS::Model (NUMBER(12,0))	Edit
SBCG_NONTERMINAL_ID		TESS::SbcgNonterminal (NUMBER(12,0))	Edit
SBCG_STREAM_ID		TESS::SbcgStream (NUMBER(12,0))	Edit
PARENT_SBCG_RECOGNITION_ID		TESS::SbcgRecognition (NUMBER(12,0))	Edit
VALUE		STRING(255)	String to match. Edit
MISMATCHES_ALLOWED		NUMBER	Number of allowable mismatches. This is not yet supported by the parser or `Reader` format. Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> TESS::SbcgRecogWeightMatrix (subclass of TESS::SbcgRecognition)

This table stores the fixed-size information about positional weight matrice as recognition primitives in a SBCG. For the details, look in the set of `TESS::Multinomial*` tables.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
SBCG_RECOGNITION_ID	no	NUMBER(12,0)	Primary key. Edit
SUBCLASS_VIEW	no	STRING(32)	Edit
MODEL_ID	no	TESS::Model (NUMBER(12,0))	Edit
SBCG_NONTERMINAL_ID		TESS::SbcgNonterminal (NUMBER(12,0))	Edit
SBCG_STREAM_ID		TESS::SbcgStream (NUMBER(12,0))	Edit
PARENT_SBCG_RECOGNITION_ID		TESS::SbcgRecognition (NUMBER(12,0))	Edit
LENGTH		NUMBER	The number of positions in the weight matrix. Edit
WIDTH		NUMBER	The width, i.e., size of the alphabet, of the model. Perhaps this should be maximum width? Edit
TOTAL_INFORMATION		FLOAT	The total information content in all positions. Edit
AVERAGE_INFORMATION		FLOAT	The average information content in each position. Edit
MAXIMUM_OBSERVATIONS		FLOAT	The number of observations in best observed position. Edit
MINIMUM_OBSERVATIONS		FLOAT	The number of observations in least observed position. Edit

Child tables:

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> TESS::SbcgRhsTerm

This table contains righthand side terms for a production. The actual recognition is kept in a view on `RecognitionImp`.

column	nulls?	type	description
SBCG_RHS_TERM_ID	no	NUMBER(12,0)	Primary key. Edit
SBCG_PRODUCTION_ID	no	TESS::SbcgProduction (NUMBER(12,0))	A link to the production containing this term. Edit
COMMENT_STRING		CLOB	This is a comment attached to the term. Edit
ORDINAL	no	NUMBER(6,0)	The position of this term in the production counted from 1. Edit
SBCG_RECOGNITION_ID	no	TESS::SbcgRecognition (NUMBER(12,0))	A link to the `Recognition` that actually does the bonuding. Edit
COUNT_MIN		NUMBER(6,0)	The minimum repeat count for this term. A null value should be taken as 1 and the `count_max` ignored. Edit
COUNT_MAX		NUMBER(6,0)	The maximum repeat count for this term. A null value should be interpreted as unbounded. Edit
COUNT_PROBABILITY		FLOAT	The probability of an additional instance when the count can vary. Can be left null when the count is fixed. Edit
CARDINALITY		NUMBER(6,0)	The required cardinality of appearences of this term. Is only used for bag productions. Edit
CAN_BE_NORMAL	no	NUMBER(1,0)	Can the RHS appear in normal sense? Edit
CAN_BE_REVERSED	no	NUMBER(1,0)	Can the RHS appear in reversed sense? Edit
CAN_OVERLAP	no	NUMBER(1,0)	Can the RHS overlap with the instance of the next RHS term? Edit
SBCG_ANNOTATION_GUIDE_ID		TESS::SbcgAnnotationGuide (NUMBER(12,0))	This is a link to an annotation guide for the RHS term. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

GUS Schema >> TESS::SbcgStream

This table contains descriptions of the alternate streams and their plugin parameters for a grammar.

column	nulls?	type	description
SBCG_STREAM_ID	no	NUMBER(12,0)	Primary key. Edit
MODEL_ID	no	TESS::Model (NUMBER(12,0))	Edit
SOURCE_CODE	no	CLOB	This field contains the source code for the stream definition. Edit
COMMENT_STRING		CLOB	This is a comment attached to the stream. Edit
STREAM_NAME	no	STRING(255)	This is the name of the stream. Edit
PLUGIN_NAME	no	STRING(255)	This is the name of the plugin that manages the stream. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Bounded Collection Grammars
TESS::SbcgRecognition
TESS::SbcgStreamParameter

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> TESS::SbcgStreamParameter

This table stores stream plug-in initialization parameters. It is just key.typed-value pairs. We just store types right here as strings.

This might be expanded to something like the `Core::Algorithm` mechanism, but noe now.

column	nulls?	type	description
SBCG_STREAM_Parameter_ID	no	NUMBER(12,0)	Primary key. Edit
SBCG_STREAM_ID	no	TESS::SbcgStream (NUMBER(12,0))	Parameter belongs to this stream (plugin). Edit
Parameter_NAME	no	STRING(255)	This the name of the parameter Edit
TYPE_NAME	no	STRING(32)	This is the parameter value's type. Edit
VALUE	no	STRING(4000)	This is a string form of the parameter value. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> RAD::ArrayDesign

column	nulls?	type	description
ARRAY_DESIGN_ID	no	NUMBER(4,0)	Edit
MANUFACTURER_ID	no	SRes::Contact (NUMBER(12,0))	Edit
TECHNOLOGY_TYPE_ID	no	Study::OntologyEntry (NUMBER(10,0))	Edit
SUBSTRATE_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
PROTOCOL_ID		RAD::Protocol (NUMBER(12,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	Edit
SOURCE_ID		STRING(100)	Edit
NAME	no	STRING(100)	Edit
VERSION		STRING(50)	Edit
DESCRIPTION		STRING(2000)	Edit
ARRAY_DIMENSIONS		STRING(50)	Edit
ELEMENT_DIMENSIONS		STRING(50)	Edit
NUMBER_OF_ELEMENTS		NUMBER(10,0)	Edit
NUM_ARRAY_COLUMNS		NUMBER(3,0)	Edit
NUM_ARRAY_ROWS		NUMBER(3,0)	Edit
NUM_GRID_COLUMNS		NUMBER(3,0)	Edit
NUM_GRID_ROWS		NUMBER(3,0)	Edit
NUM_SUB_COLUMNS		NUMBER(6,0)	Edit
NUM_SUB_ROWS		NUMBER(6,0)	Edit
SURFACE_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Platform
    RAD::ArrayDesignAnnotation
    RAD::ArrayDesignGroupLink
    RAD::CompositeElement
    RAD::Element
Assay
    RAD::Assay

Subclasses:

GUS Schema >> **RAD::ArrayDesignAnnotation**

column	nulls?	type	description
ARRAY_DESIGN_ANNOTATION_ID	no	NUMBER(5,0)	Edit
ARRAY_DESIGN_ID	no	RAD::ArrayDesign (NUMBER(4,0)) Edit	
NAME	no	STRING(500)	Edit
VALUE	no	STRING(100)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> RAD::ArrayDesignGroup

column	nulls?	type	description
ARRAY_DESIGN_GROUP_ID	no	NUMBER(8,0)	Edit
NAME	no	STRING(100)	Edit
DESCRIPTION		STRING(500)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Platform
    [RAD::ArrayDesignGroupLink](#)

Subclasses:

---

webmaster@gusdb.org

gusdb.org | Schema Browser

GUS Schema >> RAD::ArrayDesignGroupLink

column	nulls?	type	description
ARRAY_DESIGN_GROUP_LINK_ID	no	NUMBER(10,0)	Edit
ARRAY_DESIGN_GROUP_ID	no	RAD::ArrayDesignGroup (NUMBER(8,0))	Edit
ARRAY_DESIGN_ID	no	RAD::ArrayDesign (NUMBER(5,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::CompositeElement

Holds information on groups of elements.

column	nulls?	type	description
COMPOSITE_ELEMENT_ID	no	NUMBER(10,0)	primary key Edit
PARENT_ID		RAD::CompositeElement (NUMBER(10,0))	pointer back to CompositeElement Edit
ARRAY_DESIGN_ID	no	RAD::ArrayDesign (NUMBER(4,0))	Edit
SUBCLASS_VIEW	no	STRING(27)	name of the view Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	FK to SRES.ExternalDatabaseRelease. Identifies the version of the external database. Edit
SOURCE_ID		STRING(50)	unique identifier of the entry in the external database Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Platform
RAD::CompositeElement
RAD::CompositeElementAnnotation
RAD::Element
Quantified Data
RAD::CompositeElementResult

Subclasses:

RAD::SpotFamily
RAD::ShortOligoFamily
RAD::SAGETag

webmaster@gusdb.org

GUS Schema >> RAD::CompositeElementAnnotation

Overflow table to capture unstructured information about the entries in the CompositeElement table/views.

column	nulls?	type	description
COMPOSITE_ELEMENT_ANNOT_ID	no	NUMBER(12,0)	Primary Key Edit
COMPOSITE_ELEMENT_ID	no	RAD::CompositeElement (NUMBER(10,0))	Foreign Key to Edit
NAME	no	STRING(100)	text descriptor for the entry Edit
VALUE	no	STRING(500)	number value associated with the descriptor for type Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::Control

Designates elements on an array that represent controls, such as empty spots or spots with genomic DNA.

column	nulls?	type	description
CONTROL_ID	no	NUMBER(5,0)	primary key Edit
CONTROL_TYPE_ID	no	Study::OntologyEntry (NUMBER(10,0))	FK to RAD.OntologyEntry for the type of the control this is (e.g. control_spike_calibration). Edit
TABLE_ID	no	Core::TableInfo (NUMBER(5,0))	FK to Core.TableInfo the Element table that the control refers to. One of the views of RAD.ElementImp or RAD.CompositeElementImp. Edit
ROW_ID	no	NUMBER(12,0)	The primary key id for the table refered by table_id Edit
NAME		STRING(100)	text descriptor for the entry Edit
VALUE		STRING(255)	A default value for all hybridizations done on this Array, if applicable. Edit
UNIT_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	FK to RAD.OntologyEntry for the unit type of the value given. For example, concentration units. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Assay
    RAD::AssayControl

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> RAD::Element

Holds information about the element

column	nulls?	type	description
ELEMENT_ID	no	NUMBER(10,0)	primary key Edit
COMPOSITE_ELEMENT_ID		RAD::CompositeElement (NUMBER(10,0))	FK to CompositeElement Edit
ARRAY_DESIGN_ID	no	RAD::ArrayDesign (NUMBER(4,0))	Edit
DESIGN_ELEMENT_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	FK to SRES.ExternalDatabaseRelease. Identifies the version of the external database. Edit
SOURCE_ID		STRING(50)	unique identifer of the entry in the external database Edit
SUBCLASS_VIEW	no	STRING(27)	name of the view Edit
PHYSICAL_BIOSEQUENCE_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
POLYMER_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Platform
    RAD::ElementAnnotation
Quantified Data
    RAD::ElementResult

Subclasses:

RAD::ShortOligo
RAD::SAGETagMapping
RAD::Spot

---

webmaster@gusdb.org

GUS Schema >> RAD::ElementAnnotation

Overflow table to capture unstructured information about the elements.

column	nulls?	type	description
ELEMENT_ANNOTATION_ID	no	NUMBER(10,0)	Primary Key Edit
ELEMENT_ID	no	RAD::Element (NUMBER(10,0))	Foreign key to Element Edit
NAME	no	STRING(100)	text descriptor for the entry Edit
VALUE	no	STRING(500)	number value associated with the descriptor for type Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> PROT::Feature

column	nulls?	type	description
Feature_ID	no	NUMBER(12,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
Feature_SET_ID		PROT::FeatureSet (NUMBER(12,0))	Edit
ELEMENT_TYPE_ID		NUMBER(12,0)	Edit
EXTERNAL_DATABASE_RELEASE_ID		NUMBER(12,0)	Edit
SOURCE_ID		STRING(50)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Platform
   PROT::FeatureInput
Assay
   PROT::AssayProduct

Subclasses:

PROT::MALDISpot

webmaster@gusdb.org

GUS Schema >> PROT::FeatureInput

column	nulls?	type	description
Feature_INPUT_ID	no	NUMBER(8,0)	Edit
ACQUISITION_ID		PROT::Acquisition (NUMBER(12,0))	Edit
Feature_ID		PROT::Feature (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> PROT::FeatureSet

column	nulls?	type	description
Feature_SET_ID	no	NUMBER(12,0)	Edit
SUBCLASS_VIEW	no	STRING(40)	Edit
SURFACE_TYPE_ID	no	Study::OntologyEntry (NUMBER(12,0))	Edit
TECHNOLOGY_TYPE_ID		Study::OntologyEntry (NUMBER(12,0))	Edit
NAME		STRING(50)	Edit
DESCRIPTION		STRING(500)	Edit
PROTOCOL_INSTANCE_ID	no	PROT::ParameterizableInstance (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Platform
   PROT::Feature
   PROT::FSGAnnotation
Assay
   PROT::Assay

Subclasses:

PROT::MALDIPlate

GUS Schema >> PROT::FSGAnnotation

column	nulls?	type	description
FSG_ANNOTATION_ID	no	NUMBER(8,0)	Edit
Feature_SET_ID	no	PROT::FeatureSet (NUMBER(12,0))	Edit
NAME	no	STRING(50)	Edit
VALUE	no	STRING(500)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> PROT::MALDIPlate (subclass of PROT::FeatureSet)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
Feature_SET_ID	no	NUMBER(12,0)	Edit
SUBCLASS_VIEW	no	STRING(40)	Edit
SURFACE_TYPE_ID	no	Study::OntologyEntry (NUMBER(12,0))	Edit
TECHNOLOGY_TYPE_ID		Study::OntologyEntry (NUMBER(12,0))	Edit
NAME		STRING(50)	Edit
DESCRIPTION		STRING(500)	Edit
PROTOCOL_INSTANCE_ID	no	PROT::ParameterizableInstance (NUMBER(12,0))	Edit
PLATE_IDENTIFIER		STRING(100)	Edit
NUM_SPOT_X		NUMBER(12,0)	Edit
NUM_SPOT_Y		NUMBER(12,0)	Edit
MANUFACTURER_ID		NUMBER(12,0)	Edit
FIRST_SPOT_ID		NUMBER(12,0)	Edit
SECOND_SPOT_ID		NUMBER(12,0)	Edit
MODEL_TYPE_ID		NUMBER(12,0)	Edit
SPOTTING_PATTERN_TYPE_ID		NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> PROT::MALDISpot (subclass of PROT::Feature)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
Feature_ID	no	NUMBER(12,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
Feature_SET_ID		PROT::FeatureSet (NUMBER(12,0))	Edit
ELEMENT_TYPE_ID		NUMBER(12,0)	Edit
EXTERNAL_DATABASE_RELEASE_ID		NUMBER(12,0)	Edit
SOURCE_ID		STRING(50)	Edit
SPOT_IDENTIFIER		STRING(100)	Edit
X_POSITION		NUMBER(12,5)	Edit
Y_POSITION		NUMBER(12,5)	Edit
DIAMETER		STRING(100)	Edit
MATRIX_TYPE_ID		NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::SAGETag (subclass of RAD::CompositeElement)

view of CompositeElementImp for SAGE tags

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
COMPOSITE_ELEMENT_ID	no	NUMBER(10,0)	primary key Edit
PARENT_ID		RAD::CompositeElement (NUMBER(10,0))	pointer back to CompositeElement Edit
ARRAY_DESIGN_ID	no	RAD::ArrayDesign (NUMBER(4,0))	Edit
SUBCLASS_VIEW	no	STRING(27)	name of the view Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	FK to SRES.ExternalDatabaseRelease. Identifies the version of the external database. Edit
SOURCE_ID		STRING(50)	unique identifier of the entry in the external database Edit
TAG		STRING(50)	the sequence of the oligonucleotide SAGE tag Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::SAGETagMapping (subclass of RAD::Element)

view of ElementImp for mapping tags to genomic sequence

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ELEMENT_ID	no	NUMBER(10,0)	primary key Edit
COMPOSITE_ELEMENT_ID		RAD::CompositeElement (NUMBER(10,0))	FK to CompositeElement Edit
ARRAY_DESIGN_ID	no	RAD::ArrayDesign (NUMBER(4,0))	Edit
DESIGN_ELEMENT_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	FK to SRES.ExternalDatabaseRelease. Identifies the version of the external database. Edit
SOURCE_ID		STRING(50)	unique identifier of the entry in the external database Edit
SUBCLASS_VIEW	no	STRING(27)	name of the view Edit
PHYSICAL_BIOSEQUENCE_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
POLYMER_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::ShortOligo (subclass of RAD::Element)

View of ElementImp. View gives co-ordinate information for oligos on arrays manufactured using photo lithography.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ELEMENT_ID	no	NUMBER(10,0)	primary key Edit
COMPOSITE_ELEMENT_ID		RAD::CompositeElement (NUMBER(10,0))	FK to CompositeElement Edit
ARRAY_DESIGN_ID	no	RAD::ArrayDesign (NUMBER(4,0))	Edit
DESIGN_ELEMENT_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	FK to SRES.ExternalDatabaseRelease. Identifies the version of the external database. Edit
SOURCE_ID		STRING(50)	unique identifier of the entry in the external database Edit
SUBCLASS_VIEW	no	STRING(27)	name of the view Edit
PHYSICAL_BIOSEQUENCE_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
POLYMER_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
NAME		STRING(500)	text descriptor for the entry Edit
MATCH		NUMBER(3,0)	indicates whether the element is the match or mis-match probe (0 for mismatch, 1 for match) Edit
X_POSITION		STRING(50)	the position of the element in the array grid for the x-axis Edit
Y_POSITION		STRING(50)	the position of the element in the array grid for the y-axis Edit
SEQUENCE		STRING(100)	the sequence of the oligo Edit
DESCRIPTION		STRING(500)	description of the entry Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> RAD::ShortOligoFamily (subclass of RAD::CompositeElement)

View of CompositeElement. View gives common attributes for groups of oligos that map to the same gene.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
COMPOSITE_ELEMENT_ID	no	NUMBER(10,0)	primary key Edit
PARENT_ID		RAD::CompositeElement (NUMBER(10,0))	pointer back to CompositeElement Edit
ARRAY_DESIGN_ID	no	RAD::ArrayDesign (NUMBER(4,0))	Edit
SUBCLASS_VIEW	no	STRING(27)	name of the view Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	FK to SRES.ExternalDatabaseRelease. Identifies the version of the external database. Edit
SOURCE_ID		STRING(50)	unique identifier of the entry in the external database Edit
NAME		STRING(100)	text descriptor for the entry Edit
DESCRIPTION		STRING(500)	description of the entry Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::Spot (subclass of RAD::Element)

View of ElementImp. View gives coordinate information for elements on a spotted array.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ELEMENT_ID	no	NUMBER(10,0)	primary key Edit
COMPOSITE_ELEMENT_ID		RAD::CompositeElement (NUMBER(10,0))	FK to CompositeElement Edit
ARRAY_DESIGN_ID	no	RAD::ArrayDesign (NUMBER(4,0))	Edit
DESIGN_ELEMENT_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	FK to SRES.ExternalDatabaseRelease. Identifies the version of the external database. Edit
SOURCE_ID		STRING(50)	unique identifier of the entry in the external database Edit
SUBCLASS_VIEW	no	STRING(27)	name of the view Edit
PHYSICAL_BIOSEQUENCE_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
POLYMER_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
ARRAY_ROW		STRING(5)	Subarray row position for an element. Usually this is 1. Edit
ARRAY_COLUMN		STRING(5)	Subarray column position for an element. Usually this is 1. Edit
GRID_ROW		STRING(5)	Grid row position for an element. This is also known as Meta-row position. Edit
GRID_COLUMN		STRING(5)	Grid column position for an element. This is also known as Meta-column position. Edit
SUB_ROW		STRING(5)	Row position of the element within the grid. Edit
SUB_COLUMN		STRING(5)	Column position of the element within the grid. Edit
SEQUENCE_VERIFIED		NUMBER(3,0)	Flag to determine if the clone has been sequence verified. "1" if true, "0" if false. Edit
NAME		STRING(500)	text descriptor for the entry Edit
DESCRIPTION		STRING(500)	text description of the entry Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> RAD::SpotFamily (subclass of RAD::CompositeElement)

View of CompositeElement. View gives common attributes for groups of elements from same source plate/well.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
COMPOSITE_ELEMENT_ID	no	NUMBER(10,0)	primary key Edit
PARENT_ID		RAD::CompositeElement (NUMBER(10,0))	pointer back to CompositeElement Edit
ARRAY_DESIGN_ID	no	RAD::ArrayDesign (NUMBER(4,0))	Edit
SUBCLASS_VIEW	no	STRING(27)	name of the view Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	FK to SRES.ExternalDatabaseRelease. Identifies the version of the external database. Edit
SOURCE_ID		STRING(50)	unique identifier of the entry in the external database Edit
PLATE_NAME		STRING(100)	Unique identifier of source plate. Edit
WELL_LOCATION		STRING(100)	Unique identifier of well on source plate. Edit
PCR_FAILURE_FLAG		NUMBER(3,0)	Flag to determine if the PCR reaction has failed. null=Not Applicable, 0=passed PCR, 1=failed PCR, 2=multiple bands. If the distinction between 1 and 2 is not provided, designate as 1. Edit
NAME		STRING(500)	text descriptor for the entry Edit
DESCRIPTION		STRING(500)	text description of the entry Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> PROT::Acquisition

column	nulls?	type	description
ACQUISITION_ID	no	NUMBER(12,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
ASSAY_ID	no	PROT::Assay (NUMBER(12,0))	Edit
PROTOCOL_INSTANCE_ID	no	PROT::ParameterizableInstance (NUMBER(12,0))	Edit
ACQUISITION_DATE		DATE	Edit
NAME		STRING(100)	Edit
URI		STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Platform
  PROT::FeatureInput
Assay
  PROT::AcquisitionChain
  PROT::AcquisitionChannel
  PROT::AcquisitionQuantification

Subclasses:

PROT::MZScan

webmaster@gusdb.org

[GUS Schema](#) >> PROT::AcquisitionChain

column	nulls?	type	description
ACQUISITION_CHAIN_ID	no	NUMBER(12,0)	[Edit](#)
PARENT_ACQUISITION_ID	no	[PROT::Acquisition](#) (NUMBER(12,0))	[Edit](#)
CHILD_ACQUISITION_ID	no	[PROT::Acquisition](#) (NUMBER(12,0))	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

Subclasses:

GUS Schema >> PROT::AcquisitionChannel

column	nulls?	type	description
ACQUISITION_CHANNEL_ID	no	NUMBER(12,0)	Edit
ACQUISITION_ID	no	PROT::Acquisition (NUMBER(12,0))	Edit
ONTOLOGY_ENTRY_ID	no	Study::OntologyEntry (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> RAD::AcquisitionParam

Holds the name and value of the parameter for the scanner software and hardware.

column	nulls?	type	description
ACQUISITION_PARAM_ID	no	NUMBER(5,0)	primary key Edit
ACQUISITION_ID	no	RAD::Acquisition (NUMBER(8,0))	FK to Acquisition Edit
PROTOCOL_PARAM_ID		RAD::ProtocolParam (NUMBER(10,0))	Edit
NAME		STRING(100)	The name of the parameter (i.e. scanner, pmt, focal point, etc..) Edit
VALUE	no	STRING(100)	The value of the parameter, need not be numeric. e.g. (name=scanner, value=GenePixPro 4.0) or (name=pmt, value=.65), etc.. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> PROT::AcquisitionQuantification

column	nulls?	type	description
ACQUISITION_QUANTIFICATION_ID	no	NUMBER(8,0)	Edit
ACQUISITION_ID		PROT::Acquisition (NUMBER(12,0))	Edit
QUANTIFICATION_ID		PROT::Quantification (NUMBER(8,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> PROT::Assay

column	nulls?	type	description
ASSAY_ID	no	NUMBER(12,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
PROTOCOL_INSTANCE_ID	no	NUMBER(12,0)	Edit
OPERATOR_ID	no	NUMBER(12,0)	Edit
ASSAY_TYPE_ID		Study::OntologyEntry (NUMBER(12,0))	Edit
Feature_SET_ID		PROT::FeatureSet (NUMBER(12,0))	Edit
FS_IDENTIFIER		STRING(100)	Edit
ASSAY_DATE	no	DATE	Edit
EXTERNAL_DATABASE_RELEASE_ID		NUMBER(12,0)	Edit
SOURCE_ID		STRING(50)	Edit
NAME		STRING(100)	Edit
DESCRIPTION		STRING(500)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Assay
  PROT::Acquisition
  PROT::AssayChain
  PROT::AssayInput
  PROT::AssayProduct
Experimental Design
  PROT::StudyFactorValue

Subclasses:

PROT::MZRun

webmaster@gusdb.org

GUS Schema >> PROT::AssayAnnotation

column	nulls?	type	description
ASSAY_ANNOTATION_ID	no	NUMBER(12,0)	Edit
ASSAY_ID	no	NUMBER(12,0)	Edit
NAME	no	STRING(500)	Edit
VALUE	no	STRING(100)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> PROT::AssayChain

column	nulls?	type	description
ASSAY_CHAIN_ID	no	NUMBER(8,0)	Edit
SOURCE_ASSAY_ID		PROT::Assay (NUMBER(12,0))	Edit
TARGET_ASSAY_ID		PROT::Assay (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

[GUS Schema]() >> RAD::AssayControl

Relates the Control elements of an Array to Assay-specific values. For example, a spiked-in control would have a concentration associated with that particular hybridization.

column	nulls?	type	description
ASSAY_CONTROL_ID	no	NUMBER(8,0)	primary key Edit
ASSAY_ID	no	RAD::Assay (NUMBER(8,0))	FK to RAD.Assay Edit
CONTROL_ID	no	RAD::Control (NUMBER(5,0))	FK to RAD.Control, signifying which Array control the value column refers to. Edit
VALUE		STRING(255)	An assay-dependent value for the referenced Control. Edit
UNIT_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	FK to RAD.OntologyEntry for the unit type of the value given. For example milli-liters. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

[webmaster@gusdb.org]()

GUS Schema >> PROT::AssayInput

column	nulls?	type	description
ASSAY_INPUT_ID	no	NUMBER(8,0)	Edit
ASSAY_ID	no	PROT::Assay (NUMBER(12,0))	Edit
BIO_MATERIAL_MEASUREMENT_ID	no	PROT::BioMaterialMeasurement (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

GUS Schema >> RAD::AssayParam

This table holds the specific parameter setting in the application of a hybridization protocol for a given assay.

column	nulls?	type	description
ASSAY_PARAM_ID	no	NUMBER(10,0)	Primary key. Edit
ASSAY_ID	no	RAD::Assay (NUMBER(8,0))	FK to Assay. A specific assay to which this entry refers. Edit
PROTOCOL_PARAM_ID	no	RAD::ProtocolParam (NUMBER(10,0))	FK to ProtocolParam. The id of the parameter the entry refers to. Edit
VALUE	no	STRING(100)	The actual value of the parameter, which typically differs from the default value stored in the ProtocolParam table. . Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> PROT::AssayParentFile

column	nulls?	type	description
ASSAY_PARENT_FILE_ID	no	NUMBER(12,0)	[Edit](#)
ASSAY_ID	no	NUMBER(12,0)	[Edit](#)
PARENT_FILE_ID	no	[PROT::ParentFile](#) (NUMBER(12,0))	[Edit](#)
ORDER_NUM	no	NUMBER(3,0)	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> PROT::AssayProduct

column	nulls?	type	description
ASSAY_PRODUCT_ID	no	NUMBER(8,0)	Edit
BIO_MATERIAL_MEASUREMENT_ID	no	PROT::BioMaterialMeasurement (NUMBER(10,0))	Edit
ASSAY_ID	no	PROT::Assay (NUMBER(12,0))	Edit
Feature_ID		PROT::Feature (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> PROT::MZRun (subclass of PROT::Assay)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ASSAY_ID	no	NUMBER(12,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
PROTOCOL_INSTANCE_ID	no	NUMBER(12,0)	Edit
OPERATOR_ID	no	NUMBER(12,0)	Edit
ASSAY_TYPE_ID		Study::OntologyEntry (NUMBER(12,0))	Edit
Feature_SET_ID		PROT::FeatureSet (NUMBER(12,0))	Edit
FS_IDENTIFIER		STRING(100)	Edit
ASSAY_DATE	no	DATE	Edit
EXTERNAL_DATABASE_RELEASE_ID		NUMBER(12,0)	Edit
SOURCE_ID		STRING(50)	Edit
NAME		STRING(100)	Edit
DESCRIPTION		STRING(500)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> PROT::MZScan (subclass of PROT::Acquisition)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ACQUISITION_ID	no	NUMBER(12,0)	Edit
SUBCLASS_VIEW	no	STRING(30)	Edit
ASSAY_ID	no	PROT::Assay (NUMBER(12,0))	Edit
PROTOCOL_INSTANCE_ID	no	PROT::ParameterizableInstance (NUMBER(12,0))	Edit
ACQUISITION_DATE		DATE	Edit
NAME		STRING(100)	Edit
URI		STRING(255)	Edit
SCAN_NUM		NUMBER(12,0)	Edit
MSLEVL		NUMBER(12,0)	Edit
PEAK_COUNT		NUMBER(12,0)	Edit
POLARITY		STRING(100)	Edit
SCAN_TYPE		STRING(100)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> PROT::ParentFile

column	nulls?	type	description
PARENT_FILE_ID	no	NUMBER(12,0)	Edit
PARENT_FILE_TYPE_ID	no	Study::OntologyEntry (NUMBER(12,0))	Edit
URI	no	STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Assay
   PROT::AssayParentFile

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> PROT::Quantification

column	nulls?	type	description
QUANTIFICATION_ID	no	NUMBER(8,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
PROTOCOL_INSTANCE_ID		PROT::ParameterizableInstance (NUMBER(10,0))	Edit
QUANTIFICATION_DATE		DATE	Edit
NAME		STRING(100)	Edit
URI		STRING(255)	Edit
DESCRIPTION		STRING(500)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Data Transformations and Analyses
    PROT::DataTransformation
Assay
    PROT::AcquisitionQuantification
Quantified Data
    PROT::ElementResult

Subclasses:

PROT::MZPeak

GUS Schema >> RAD::QuantificationParam

Holds information on the parameter settings for a given quantification, including software and version. The name of each parameter and its value are held in this table.

column	nulls?	type	description
QUANTIFICATION_PARAM_ID	no	NUMBER(5,0)	primary key Edit
QUANTIFICATION_ID	no	RAD::Quantification (NUMBER(8,0))	FK to Quantification. The quantification to which this parameter refers. Edit
PROTOCOL_PARAM_ID		RAD::ProtocolParam (NUMBER(10,0))	Edit
NAME		STRING(100)	text descriptor for the entry Edit
VALUE	no	STRING(100)	number value associated with the descriptor for type Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> RAD::RelatedAcquisition

Relations between acquisitions are stored here, for example the red and green channels of a two channel microarray. Each acquisition is connected by this table to any it is related to. So for example a 2-channel microarray would require two entries, each channel being the acquisition and associated acquisition in turn. A 3-channel array would require six entries. Etc.

column	nulls?	type	description
RELATED_ACQUISITION_ID	no	NUMBER(4,0)	primary key Edit
ACQUISITION_ID	no	RAD::Acquisition (NUMBER(8,0))	FK to Acquisition, pointing to the primary acquisition for this entry. Edit
ASSOCIATED_ACQUISITION_ID	no	RAD::Acquisition (NUMBER(8,0))	FK to the acquistion ID of the related acquisition for this entry. Edit
NAME		STRING(100)	Each relation can be given a name for user reference. Edit
DESIGNATION		STRING(50)	The designation, if any, of the main acquisition. E.g. red dye or green dye, etc.. Edit
ASSOCIATED_DESIGNATION		STRING(50)	The designation, if any, of the associated acquisition. E.g. red dye or green die, etc.. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::RelatedQuantification

Used for multi-channel arrays to retrieve, for a given quantification, the quantification corresponding to the other channel(s) in the competitive hybridization.

column	nulls?	type	description
RELATED_QUANTIFICATION_ID	no	NUMBER(4,0)	primary key Edit
QUANTIFICATION_ID	no	RAD::Quantification (NUMBER(8,0))	FK to Quantification. The quantification_id for the given channel. Edit
ASSOCIATED_QUANTIFICATION_ID	no	RAD::Quantification (NUMBER(8,0))	FK to Quantification. The quantification_id for the co-hybridized assay to which this entry refers. Edit
NAME		STRING(100)	text descriptor for the entry Edit
DESIGNATION		STRING(50)	Channel corresponding to the given quantification. Edit
ASSOCIATED_DESIGNATION		STRING(50)	Channel corresponding to the co-hybridized assay to which this entry refers. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::AffymetrixCEL (subclass of RAD::ElementResult)

View of ElementResultImp to hold array data quantified using software by Affymetrix.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ELEMENT_RESULT_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
ELEMENT_ID	no	RAD::Element (NUMBER(10,0))	Edit
COMPOSITE_ELEMENT_RESULT_ID		RAD::CompositeElementResult (NUMBER(10,0))	Edit
QUANTIFICATION_ID	no	RAD::Quantification (NUMBER(8,0))	Edit
MEAN		FLOAT	The 75th percentile of the pixel intensities in the stripped feature. Edit
STDV		FLOAT	The standard deviation of the pixel intensities in the stripped feature. The stripped feature contains the remaining pixels after the bordering pixels of the probe cell have been excluded. Edit
NPIXELS		NUMBER(12,0)	The number of pixels in the stripped feature. Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> RAD::ElementResult

column	nulls?	type	description
ELEMENT_RESULT_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
ELEMENT_ID	no	RAD::Element (NUMBER(10,0))	Edit
COMPOSITE_ELEMENT_RESULT_ID		RAD::CompositeElementResult (NUMBER(10,0))	Edit
QUANTIFICATION_ID	no	RAD::Quantification (NUMBER(8,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

RAD::ScanAlyzeElementResult
RAD::ArrayVisionElementResult
RAD::AgilentElementResult
RAD::GEMToolsElementResult
RAD::GenePixElementResult
RAD::QuantArrayElementResult
RAD::SpotElementResult
RAD::AffymetrixCEL

webmaster@gusdb.org

GUS Schema >> RAD::AffymetrixMAS4 (subclass of RAD::CompositeElementResult)

This view of CompositeElementResultImp is for quantification of microarray data using Affymetrix Microarray Suite, Version 4.0.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
COMPOSITE_ELEMENT_RESULT_ID	no	NUMBER(10,0)	primary key Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
COMPOSITE_ELEMENT_ID	no	RAD::CompositeElement (NUMBER(10,0))	FK to CompositeElement, the corresponding entry in the composite element view this result is referring to. Edit
QUANTIFICATION_ID	no	RAD::Quantification (NUMBER(8,0))	FK to quantification. There can be CompositeElementResult values without corresponding ElementResult's, as can be the case in summary values for Affy probe sets. Edit
POSITIVE_PROBE_PAIRS		NUMBER(3,0)	The number of probe pairs where the intensity of the perfect match (PM) is significantly greater than the corresponding intensity of the mismatch (MM). Edit
NEGATIVE_PROBE_PAIRS		NUMBER(3,0)	The number of probe pairs where the intensity of the mismatch (MM) is significantly greater than the corresponding intensity of the perfect match (PM). Edit
NUM_PROBE_PAIRS_USED		NUMBER(3,0)	Pairs used = total probe pairs per probe set – (probe pairs masked in a mask file) – (probe pairs masked in the image). Edit
PAIRS_IN_AVERAGE		NUMBER(5,0)	A trimmed probe set that excludes probes with extremely intense or weak signal from the analysis. Edit
LOG_AVERAGE_RATIO		FLOAT	The log of the PM/MM ratio is calculated for each probe pair in a probe set. The average is then taken across the pairs_in_average. This is a measure of the strength of the hybridization. A greater log average indicates an increased likelihood of the presence of that transcript. Edit
AVERAGE_DIFFERENCE		FLOAT	The average difference is a measure of the relative expression of a transcript and can be used to compare expression across experiments. It is calculated by taking the average of the difference between the PM – MM of each probe pair in the pairs_in_average (after excluding extreme values as described for pairs_in_average). Edit
ABSOLUTE_CALL		STRING(500)	A categorical statement about the likelihood of the presence of a transcript. The calls are: present (P), Marginal (M), or Absent (A). The call is made by a weight matrix which takes the following into consideration: the ratio of positive to negative probe pairs, the positive fraction (the number of positive probe pairs / the nuber of probe pairs used), and the log average ratio. Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::CompositeElementResult

Implementation table. Holds a summary value for the composite element view, i.e. if the composite element is a set of duplicate spots on the array, this might be the average intensity; or if the composite element is a set of affy probes in one probe set this might be the affy summary intensity for that probe set.

column	nulls?	type	description
COMPOSITE_ELEMENT_RESULT_ID	no	NUMBER(10,0)	primary key Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
COMPOSITE_ELEMENT_ID	no	RAD::CompositeElement (NUMBER(10,0))	FK to CompositeElement, the corresponding entry in the composite element view this result is referring to. Edit
QUANTIFICATION_ID	no	RAD::Quantification (NUMBER(8,0))	FK to quantification. There can be CompositeElementResult values without corresponding ElementResult's, as can be the case in summary values for Affy probe sets. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Quantified Data
   RAD::ElementResult

Subclasses:

RAD::RMAExpress
RAD::ProbeProfiler
RAD::MOIDResult
RAD::AffymetrixMAS4
RAD::AffymetrixMAS5
RAD::SAGETagResult

webmaster@gusdb.org

GUS Schema >> RAD::AffymetrixMAS5 (subclass of RAD::CompositeElementResult)

This view of CompositeElementResultImp is for quantification of microarray data using Affymetrix Microarray Suite, Version 5.0.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
COMPOSITE_ELEMENT_RESULT_ID	no	NUMBER(10,0)	primary key Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
COMPOSITE_ELEMENT_ID	no	RAD::CompositeElement (NUMBER(10,0))	FK to CompositeElement, the corresponding entry in the composite element view this result is referring to. Edit
QUANTIFICATION_ID	no	RAD::Quantification (NUMBER(8,0))	FK to quantification. There can be CompositeElementResult values without corresponding ElementResult's, as can be the case in summary values for Affy probe sets. Edit
SIGNAL		FLOAT	The signal represents the amount of transcript in solution. For each PM intensity, a matching MM probe provides a reference background hybridization intensity. If the MM value is less than the PM value, the algorithm uses the MM value directly. However, if the MM value is larger than the PM value, the algorithm creates an adjusted MM value based on the average difference intensity between log2 PM and log2 MM, or if that measure is too small, some fraction of PM. Edit
DETECTION		STRING(5)	The call in an absolute analysis indicating the presence (P), absence (A), marginal (M), or no call (NC) of a transcript. Edit
DETECTION_P_VALUE		FLOAT	Significance level of the detection call. Edit
STAT_PAIRS		NUMBER(5,0)	The number of probe pairs in each probe set. Edit
STAT_PAIRS_USED		NUMBER(5,0)	The number of probe pairs used by the Statistical Expression algorithm to determine the absolute detection call. This is determined by: (number of pairs) – (masked probe pairs) – (saturated MM probe pairs). Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::AgilentElementResult (subclass of RAD::ElementResult)

This view of ElementResultImp is for array data quantified with the Agilent Feature Extraction software.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ELEMENT_RESULT_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
ELEMENT_ID	no	RAD::Element (NUMBER(10,0))	Edit
COMPOSITE_ELEMENT_RESULT_ID		RAD::CompositeElementResult (NUMBER(10,0))	Edit
QUANTIFICATION_ID	no	RAD::Quantification (NUMBER(8,0))	Edit
MEAN_SIGNAL		FLOAT	Raw mean signal of this feature in the channel relative to this quantification (inlier pixels). Edit
BG_MEAN_SIGNAL		FLOAT	Mean local background signal for this feature in the channel relative to this quantification (inlier pixels). Edit
PIX_SDEV		FLOAT	Standard deviation of all inlier pixels of this feature in the channel relative to this quantification. Edit
BG_PIX_SDEV		FLOAT	Standard deviation of all inlier pixels per local background of this feature in the channel relative to this quantification. Edit
MEDIAN_SIGNAL		FLOAT	Raw median signal of this feature in the channel relative to this quantification (inlier pixels). Edit
BG_MEDIAN_SIGNAL		FLOAT	Median local background signal for this feature in the channel relative to this quantification (inlier pixels). Edit
POSITION_X		FLOAT	X coordinate of the feature centroid in pixels. Edit
POSITION_Y		FLOAT	Y coordinate of the feature centroid in pixels. Edit
SPOT_RADIUS_X		FLOAT	Radius of the feature (X-axis) when Calculate Spot Size and WholeSpot method are selected. Edit
SPOT_RADIUS_Y		FLOAT	Radius of the feature (Y-axis) when Calculate Spot Size and WholeSpot method are selected. Edit
PIX_CORRELATION		FLOAT	Ratio of estimated feature covariance in RedGreen space to product of feature Standard Deviation in RedGreen space. This is a cumulative quantitation of the tendency of pixels belonging to this feature in Red and Green spaces to co-vary. Edit
BG_PIX_CORRELATION		FLOAT	The same concept as pix_correlation but in case of background. Edit
P_VAL_FEAT_EQ_BG		FLOAT	P-value from t-test of significance between mean signal and background. Edit
IS_FOUND		NUMBER(3,0)	A boolean used to flag found (strong) features for the channel relative to this quantification. A feature is considered found if the found spot centroid is within the bounds of the spot deviation limit with respect to corresponding nominal centroid. 1=IsFound, 0=IsNotFound. Edit
IS_SATURATED		NUMBER(3,0)	Boolean flag indicating if a feature is saturated or not. A feature is saturated if 50% of its pixels are above the saturation threshold. 1=Saturated, 0=Non saturated. Edit
IS_FEAT_NON_UNIF_OL		NUMBER(3,0)	Boolean flag indicating if a feature is a NonUniformity Outlierornot. A feature is non-uniform if its pixel noise exceeds a threshold established for a "uniform" feature. 1 indicates non-uniformity outlier for the channel relative to this quantification. Edit
IS_BG_NON_UNIF_OL		NUMBER(5,0)	The same concept as is_feat_non_unif_ol but for background. Edit
IS_FEAT_POPN_OL		NUMBER(5,0)	Boolean flag indicating if a feature is a Population Outlier or not. Probes with replicate features on a

			microarray are examined using population statistics. A feature is a population outlier if its signal is less than a lower threshold or exceeds an upper threshold determined using the interquartile range of the population. 1 indicates that the feature is a population outlier for the channel relative to this quantification. Edit
IS_BG_POPN_OL		NUMBER(5,0)	The same concept as is_feat_popn_ol but for background. Edit
IS_MANUAL_FLAG		NUMBER(12,0)	Edit
IS_POS_AND_SIGNIF		NUMBER(12,0)	Boolean flag indicating if the mean signal of this feature is greater than the corresponding background (selected by user) and if this difference is significant. Significance is established via a 2-sided t-test against the user-settable maximum p-value. 1 indicates that this feature is positive and significant above background. Edit
IS_WELL_ABOVE_BG		NUMBER(12,0)	A boolean flag indicating if a feature is WellAbove Background or not. Feature passes is_pos_and_signif and additionally the background subtracted signal is greater than 2.6* bkg. SD. Edit
IS_USED_BG_ADJUST		NUMBER(12,0)	A boolean used to flag features used for computation of global BG offset. 1=Features used; 0= Feature not used. Edit
NUM_PIX_OL_HI		NUMBER(12,0)	Number of outlier pixels per feature with intensity > upper threshold set via the pixel outlier rejection method. The number is computed independently in each channel. Edit
NUM_PIX_OL_LO		NUMBER(12,0)	Number of outlier pixels per feature with intensity < lower threshold set via the pixel outlier rejection method. The number is computed independently in each channel. Edit
NUM_PIX		NUMBER(12,0)	Total number of pixels used to compute feature statistics; i.e. total number of inlier pixels/per spot; same in both channels. Edit
BG_NUM_PIX		NUMBER(12,0)	Total number of pixels used to compute local background statistics per spot; i.e. total number of background inlier pixels; same in both channels. Edit
NUM_SAT_PIX		NUMBER(12,0)	Total number of saturated pixels per feature, computed per channel. Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::ArrayVisionElementResult (subclass of RAD::ElementResult)

This view of ElementResultImp is for array data quantified with the ArrayVision software by Imaging Research Inc.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ELEMENT_RESULT_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
ELEMENT_ID	no	RAD::Element (NUMBER(10,0))	Edit
COMPOSITE_ELEMENT_RESULT_ID		RAD::CompositeElementResult (NUMBER(10,0))	Edit
QUANTIFICATION_ID	no	RAD::Quantification (NUMBER(8,0))	Edit
FOREGROUND		FLOAT	One of the ArrayVision principal measures. The possibilities are: Density, ARM Density, Median Density, MTM Density, Volume, AR Volume. The chosen measure should be specified in QuantificationParam. If the chosen measure uses a parameter (e.g. ARM Density uses one), this too should be specified in QuantificationParam. Edit
BACKGROUND		FLOAT	The background density or volume. The background density is from one of the ArrayVision background measures: Mean or Median. The chosen one should be specified in QuantificationParam. The background volume value is adjusted to the size of the spot, where applicable. Edit
SD		FLOAT	The SD measure in ArrayVision, that is the standard deviation of the pixel density values. Edit
MAD		FLOAT	The MAD measure in ArrayVision, that is the median of the absolute values of deviations from the median density. Edit
SIGNAL_TO_NOISE		FLOAT	Signal-to-Noise ratio: spot density minus background density, divided by the SD of the background density. Edit
PERCENT_REMOVED		FLOAT	Percentage of pixels excluded in the calculation of ARM density and MTM density, when these are used as principal measures. Edit
PERCENT_REPLACED		FLOAT	Percentage of pixels replaced with estimated values in the calculation of AR Volume, when this is used as principal measure. Edit
PERCENT_AT_FLOOR		FLOAT	Proportion of foreground pixels at the lowermost limit of the density scale (threshold should be specified in QuantificationParam). These pixels are clipped. Edit
PERCENT_AT_CEILING		FLOAT	Proportion of foreground pixels at the uppermost limit of the density scale (threshold should be specified in QuantificationParam). These pixels are clipped. Edit
BKG_PERCENT_AT_FLOOR		FLOAT	Proportion of background pixels at the lowermost limit of the density scale (threshold should be specified in QuantificationParam). These pixels are clipped. Edit
BKG_PERCENT_AT_CEILING		FLOAT	Proportion of background pixels at the uppermost limit of the density scale (threshold should be specified in QuantificationParam). These pixels are clipped. Edit
X		STRING(50)	Horizontal co-ordinate of this spot: value and units. Note that the (0,0) point is in the bottom left of the image. Edit
Y		STRING(50)	Vertical co-ordinate of this spot: value and units. Note that the (0,0) point is in the bottom left of the image. Edit
AREA		STRING(50)	Area of this spot. Edit
FLAG		NUMBER(3,0)	Visual flag (1 if the spot was flagged, 0 otherwise). Edit

Child tables:

Subclasses:

GUS Schema >> PROT::ElementResult

column	nulls?	type	description
ELEMENT_RESULT_ID	no	NUMBER(10,0)	Edit
QUANTIFICATION_ID	no	PROT::Quantification (NUMBER(8,0))	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

PROT::MZPeakResult

GUS Schema >> RAD::GEMToolsElementResult (subclass of RAD::ElementResult)

This view of ElementResultImp is for array data quantified with the GEMTools software by Incyte Genomics

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ELEMENT_RESULT_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
ELEMENT_ID	no	RAD::Element (NUMBER(10,0))	Edit
COMPOSITE_ELEMENT_RESULT_ID		RAD::CompositeElementResult (NUMBER(10,0))	Edit
QUANTIFICATION_ID	no	RAD::Quantification (NUMBER(8,0))	Edit
SIGNAL		FLOAT	The background subtracted intensity of the element. Edit
SIGNAL_TO_BACKGROUND		FLOAT	The average (not background subtracted) intensity of the element, divided by the average local background in the element region. Edit
AREA_PERCENTAGE		FLOAT	The size of the element used to compute the signal relative to the average element size on the GEM element area. Element area is always the same for both the Cy3 and Cy5 channel. Edit
VISUAL_FLAG		NUMBER(3,0)	Visual flag (1 if the spot was flagged, 0 otherwise). Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> RAD::GenePixElementResult (subclass of RAD::ElementResult)

Stores results from GenePix *.gpr files.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ELEMENT_RESULT_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
ELEMENT_ID	no	RAD::Element (NUMBER(10,0))	Edit
COMPOSITE_ELEMENT_RESULT_ID		RAD::CompositeElementResult (NUMBER(10,0))	Edit
QUANTIFICATION_ID	no	RAD::Quantification (NUMBER(8,0))	Edit
FOREGROUND_SD		FLOAT	The standard deviation of the intensities for pixels within a circular feature−indicator. Pixels that contact the feature−indicator circle itself are excluded. Edit
BACKGROUND_SD		FLOAT	Local or global standard deviation of the non−specific signal fluorescence. This value is subtracted from the foreground value prior to ratio calculations. Edit
SPOT_DIAMETER		FLOAT	Circular feature indicators can be variable in diameter. Edit
FOREGROUND_MEAN		FLOAT	The mean of the intensities for pixels within a circular feature−indicator. Pixels that contact the feature−indicator circle itself are excluded. Edit
FOREGROUND_MEDIAN		FLOAT	The median of the intensities for pixels within a circular feature−indicator. Pixels that contact the feature−indicator circle itself are excluded. Edit
BACKGROUND_MEAN		FLOAT	Local or global mean of the non−specific signal fluorescence. This value is subtracted from the foreground value prior to ratio calculations. Edit
BACKGROUND_MEDIAN		FLOAT	Local or global median of the non−specific signal fluorescence. This value is subtracted from the foreground value prior to ratio calculations. Edit
PERCENT_OVER_BG_PLUS_ONE_SD		FLOAT	The percentage of pixels greater than one standard deviation above background. A measure of the quality of the data in an experiment. Edit
PERCENT_OVER_BG_PLUS_TWO_SDS		FLOAT	The percentage of pixels greater than two standard deviations above background. A measure of the quality of the data in an experiment. Edit
PERCENT_FOREGROUND_SATURATED		FLOAT	Percentage of pixels that have an intensity equal to the maximum value of 65535. This indicates an inaccurate signal measurement. Adjusting the PMT gain should prevent such saturation. Edit
MEAN_OF_RATIOS		FLOAT	Arithmetic mean of pixel−by−pixel ratios of pixel intensities, with the median background subtracted. More affected by outliers than the median_of_ratios. Edit
MEDIAN_OF_RATIOS		FLOAT	Median of pixel−by−pixel ratios of pixel intensities, with the median background subtracted. More robust to outliers than the mean_of_ratios. Edit
RATIOS_SD		FLOAT	Standard deviation of pixel−pixel intensity ratios. Edit
RGN_RATIO		FLOAT	The regression ratio. Edit
RGN_R_SQUARED		FLOAT	The coefficient of determination for the current regression value. Edit
NUM_FOREGROUND_PIXELS		NUMBER(5,0)	Number of pixels used to calculate the foreground intensity. Edit
NUM_BACKGROUND_PIXELS		NUMBER(5,0)	Number of pixels used to calculate the background intensity. Edit

FLAG		NUMBER(3,0)	Visual inspection may prompt flagging of features that should be excluded from analysis. Flags are stored as numerical values as follows: good = 100, bad = -100, not found = -50, absent = -75, unflagged = 0. Advanced flagging features allow the user to specify custom flag values. Edit
AUTOFLAG		NUMBER(3,0)	Used to mark whether or not a good or bad flag was applied from the Flag. Edit
CIRCULARITY		FLOAT	Uses a metric based on the variance of the distance of each boundary pixel to the centroid of the feature: 100 is most circular, 0 is most non-circular. Circular features are always 100, square features are always 79 (p/4*100). Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::MOIDResult (subclass of RAD::CompositeElementResult)

Holds results from MOID analysis of data generated on an Affymetrix platform.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
COMPOSITE_ELEMENT_RESULT_ID	no	NUMBER(10,0)	primary key Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
COMPOSITE_ELEMENT_ID	no	RAD::CompositeElement (NUMBER(10,0))	FK to CompositeElement, the corresponding entry in the composite element view this result is referring to. Edit
QUANTIFICATION_ID	no	RAD::Quantification (NUMBER(8,0))	FK to quantification. There can be CompositeElementResult values without corresponding ElementResult's, as can be the case in summary values for Affy probe sets. Edit
EXPRESSION		FLOAT	Expression is calculated as the intensity difference between the curves S and B at the 70th accumulative percentile. S(intensity) is the integral distribution of the match signal for this probe set and B(intensity) is the non−probe−specific integral background distribution and is derived from the intensities of the 5% darkest probe sets. Edit
LOWER_BOUND		FLOAT	The left end point of an appropriate confidence interval for expression. Edit
UPPER_BOUND		FLOAT	The right end point of an appropriate confidence interval for expression Edit
LOG_P		FLOAT	The logarithm (base 10) of the probability that the observed signal distribution for this probe set could be explained by background. Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> PROT::MZPeak (subclass of PROT::Quantification)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
QUANTIFICATION_ID	no	NUMBER(8,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
PROTOCOL_INSTANCE_ID		PROT::ParameterizableInstance (NUMBER(10,0))	Edit
QUANTIFICATION_DATE		DATE	Edit
NAME		STRING(100)	Edit
URI		STRING(255)	Edit
DESCRIPTION		STRING(500)	Edit

Child tables:

Subclasses:

GUS Schema >> PROT::MZPeakResult (subclass of PROT::ElementResult)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ELEMENT_RESULT_ID	no	NUMBER(10,0)	Edit
QUANTIFICATION_ID	no	PROT::Quantification (NUMBER(8,0))	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
BASE64PEAK		CLOB	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::ProbeProfiler (subclass of RAD::CompositeElementResult)

This view of CompositeElementResultImp is for quantification of microarray data using Probe Profiler (Version 2.0).

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
COMPOSITE_ELEMENT_RESULT_ID	no	NUMBER(10,0)	primary key Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
COMPOSITE_ELEMENT_ID	no	RAD::CompositeElement (NUMBER(10,0))	FK to CompositeElement, the corresponding entry in the composite element view this result is referring to. Edit
QUANTIFICATION_ID	no	RAD::Quantification (NUMBER(8,0))	FK to quantification. There can be CompositeElementResult values without corresponding ElementResult's, as can be the case in summary values for Affy probe sets. Edit
E_SCORE		FLOAT	Estimate of transcript abundance for this probeset. Edit
SD		FLOAT	Estimate of the standard deviation for the e_score assuming additive variance. Value is intended to be reflective of background noise in the low signal intensity ranges. Edit
SD_LOG		FLOAT	Estimate of the natural log of the multiplicative standard deviation assuming multiplicative variance. This variance is used if expression score is large enough. Specifically, if the multiplicative variance is greater than the noise_threshold_ratio times the additive variance. If not used in estimating, sd_log is set equal to zero. Edit
P_VALUE		FLOAT	The p-value testing the null hypothesis that this probeset has an expression score (e_score) greater than zero. Assumes additive variance ONLY. Edit
PP_USED		NUMBER(5,0)	Number of probe pairs used to calculate expression score (e_score). Edit
PP_NOT_USED		NUMBER(5,0)	Number of probe pairs NOT used to calculate expression score (e_score). Edit
PP_SATURATED		NUMBER(5,0)	Number of probe pairs that were saturated. Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> RAD::QuantArrayElementResult (subclass of RAD::ElementResult)

This view of ElementResultImp is for array data quantified with the Perkin Elmer QuantArray software.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ELEMENT_RESULT_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
ELEMENT_ID	no	RAD::Element (NUMBER(10,0))	Edit
COMPOSITE_ELEMENT_RESULT_ID		RAD::CompositeElementResult (NUMBER(10,0))	Edit
QUANTIFICATION_ID	no	RAD::Quantification (NUMBER(8,0))	Edit
INTENSITY		FLOAT	The foreground signal of this spot. Edit
BACKGROUND		FLOAT	The background signal for this spot. Edit
INTENSITY_SDEV		FLOAT	Standard deviation of the foreground pixel intensity. Edit
BACKGROUND_SDEV		FLOAT	Standard deviation of the background pixel intensity. Edit
X_LOCATION		FLOAT	x coordinate of the spot centroid in pixels. Edit
Y_LOCATION		FLOAT	y coordinate of the spot centroid in pixels. Edit
DIAMETER		FLOAT	The average diameter of this spot. This is useful to determine whether the spot is close to the expected size or is perhaps an artifact on the slide. Nominal diameter D is set in the Array Pattern dialog box. Edit
AREA		FLOAT	The area of the spot as the number of pixels used when calculating spot intensity. This is useful to determine whether the spot is close to the expected size or is an artifact, or has a satellite artifact on the slide. Artifacts that are connected to a spot can be recognized by a larger than expected area. The Nominal value of the area is pi*D^2/4, where D is the nominal diameter. Edit
FOOTPRINT		FLOAT	The distance between the expected position of a spot and its actual measured position. This is useful to determine the overall print quality of the microarray. This measurement also enables researchers to distinguish spots from artifacts on the slide. Edit
CIRCULARITY		FLOAT	The degree to which the spot matches a perfect circle. This is useful to determine the print quality of the microarray. Edit
SPOT_UNIFORMITY		FLOAT	The uniformity of the pixels used in the spot intensity calculation. This helps in determining the quality of the spot and aids in selecting the correct quantification output method (mean, median, total). A large uniformity value corresponds to a large distribution of pixels within the spot. In some cases, this may cause a mean or median value to fluctuate causing poor repeatability. A total quantification value may give a better choice in this case. Edit
BKG_UNIFORMITY		FLOAT	The uniformity of pixels used in the local background intensity calculation. This measurement can be used to determine the quality of microarray processes such as hybridizing and washing. More accurate spot intensities can be produced with a low intensity, uniform background around each spot. Edit
SN_RATIO		FLOAT	The ratio of the spot intensity to the standard deviation of the local background of all spots in the microarray. Spots with a low signal to noise ratio can be quickly identified for later review. Edit
CONFIDENCE		FLOAT	Edit

DIAMETER_FILTER		NUMBER(3,0)	Edit
AREA_FILTER		NUMBER(3,0)	Edit
FOOTPRINT_FILTER		NUMBER(3,0)	Edit
CIRCULARITY_FILTER		NUMBER(5,0)	Edit
SPOT_UNIFORMITY_FILTER		NUMBER(5,0)	Edit
BKG_UNIFORMITY_FILTER		NUMBER(5,0)	Edit
SN_RATIO_FILTER		NUMBER(12,0)	Edit
REPLICATE_FILTER		NUMBER(12,0)	Edit
IGNORE_FILTER		NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> RAD::RMAExpress (subclass of RAD::CompositeElementResult)

This view of CompositeElementResultImp is for (Affymetrix) data quantified (and possibly processed) with RMAExpress. RMA (Robust Multichip Average) consists of three steps: a background adjustment (optional), quantile normalization (optional), and finally summarization.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
COMPOSITE_ELEMENT_RESULT_ID	no	NUMBER(10,0)	primary key Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
COMPOSITE_ELEMENT_ID	no	RAD::CompositeElement (NUMBER(10,0))	FK to CompositeElement, the corresponding entry in the composite element view this result is referring to. Edit
QUANTIFICATION_ID	no	RAD::Quantification (NUMBER(8,0))	FK to quantification. There can be CompositeElementResult values without corresponding ElementResult's, as can be the case in summary values for Affy probe sets. Edit
RMA_EXPRESSION_MEASURE		FLOAT	The expression value for this probe set computed by RMAExpress. This might or might not have been background corrected and might or might not have been normalized, as specified by the parameter settings stored in QuantificationParam. Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::SAGETagResult (subclass of RAD::CompositeElementResult)

view of CompositeElementResultImp

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
COMPOSITE_ELEMENT_RESULT_ID	no	NUMBER(10,0)	primary key Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
COMPOSITE_ELEMENT_ID	no	RAD::CompositeElement (NUMBER(10,0))	FK to CompositeElement, the corresponding entry in the composite element view this result is referring to. Edit
QUANTIFICATION_ID	no	RAD::Quantification (NUMBER(8,0))	FK to quantification. There can be CompositeElementResult values without corresponding ElementResult's, as can be the case in summary values for Affy probe sets. Edit
TAG_COUNT		NUMBER(12,0)	the number of times a SAGE tag was found in a sequence run (i.e., assay). Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::ScanAlyzeElementResult (subclass of RAD::ElementResult)

This view of ElementResultImp is for array data quantified with the Scanalyze software.

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ELEMENT_RESULT_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
ELEMENT_ID	no	RAD::Element (NUMBER(10,0))	Edit
COMPOSITE_ELEMENT_RESULT_ID		RAD::CompositeElementResult (NUMBER(10,0))	Edit
QUANTIFICATION_ID	no	RAD::Quantification (NUMBER(8,0))	Edit
I		FLOAT	the mean pixel value within the spot. Edit
B		FLOAT	median pixel intensity in the local background region Edit
BA		FLOAT	Edit
SPIX		NUMBER(12,0)	the number of pixels in the spot feature Edit
BGPIX		NUMBER(12,0)	the number of pixels in the local background region Edit
TOP_COORDINATE		NUMBER(12,0)	Edit
LEFT_COORDINATE		NUMBER(12,0)	Edit
BOT_COORDINATE		NUMBER(12,0)	Edit
RIGHT_COORDINATE		NUMBER(12,0)	Edit
FLAG		NUMBER(3,0)	user defined spot flag (default 0) Edit
MRAT		FLOAT	contains the median of (CH2PI – CH2B)/(CH1PI – CH1B) where CH1PI and CH2PI represent single pixel intensities. Edit
REGR		FLOAT	when ch1 pixel intensities plotted against ch2 pixel intensities, regr is estimate of slope of line by simple linear regression of channel 2 on channel 1 Edit
CORR		FLOAT	the correlation between ch1 and ch2 pixels within the spot (useful for quality control – high values imply better fit and good spot quality) Edit
LFRAT		FLOAT	when ch1 pixel intensities plotted against ch2 pixel intensities, lfrat is estimate of slope of line by least–squares fit Edit
GTB1		FLOAT	fraction of pixels in the spot greater than background. Edit
GTB2		FLOAT	fraction of pixels in the spot greater than 1.5*background. Edit
EDGEA		FLOAT	mean magnitude of the horizontal and vertical Sobel edge vectors contained within each spot. Edit
KSD		FLOAT	the value of the Komogorov–Smimov statistic that assesses the likelihood that the spot pixel intensity distribution is drawn from the background distribution. The actual probabilities are stored in the ksp attribute. Edit
KSP		FLOAT	p–value from the Kolmogorov–Smirnov statistic. Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> RAD::SpotElementResult (subclass of RAD::ElementResult)

This view of ElementResultImp is for array data quantified with the Spot software by CSIRO

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
ELEMENT_RESULT_ID	no	NUMBER(10,0)	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
ELEMENT_ID	no	RAD::Element (NUMBER(10,0))	Edit
COMPOSITE_ELEMENT_RESULT_ID		RAD::CompositeElementResult (NUMBER(10,0))	Edit
QUANTIFICATION_ID	no	RAD::Quantification (NUMBER(8,0))	Edit
MEDIAN		FLOAT	The median pixel value within the spot. Edit
MORPH		FLOAT	Value of a morphological opening at the nominal spot centre. Used as a measure of background. Edit
IQR		FLOAT	The interquartile range of the logged pixel values within the spot. Edit
MEAN		FLOAT	The average pixel value within the spot. Edit
BG_MEDIAN		FLOAT	The median pixel value in the background region. Edit
BG_MEAN		FLOAT	The average pixel value in the background region. Edit
BG_SD		FLOAT	The interquartile range of the logged pixel values in the background region. Edit
VALLEY		FLOAT	Median of median image values in four neighbouring valley regions. Valley regions are square regions between spots. Edit
MORPH_ERODE		FLOAT	Value of a morphological erosion at the nominal spot centre. Edit
MORPH_CLOSE_OPEN		FLOAT	Value of a morphological closing followed by a morphological opening at the spot centre. Edit
AREA		NUMBER(12,0)	The number of pixels in the segmented spot. Edit
PERIMETER		NUMBER(12,0)	The perimeter of the spot as measured by the number of pixels in the spot which have a neighbouring pixel outside the spot. Each pixel is considered to have 8 neighbours in this process. Edit
CIRCULARITY		FLOAT	A measure of circularity of the spot equal to area times four pi divided by perimeter squared. This is ideally one for circles and more than one for other shapes. Edit
BADSPOT		NUMBER(3,0)	A flag which is 1 if the spot area is greater than the product of the horizontal and vertical average spot separations, and 0 otherwise. Edit
VISUAL_FLAG		NUMBER(3,0)	Visual flag (1 if the spot was flagged, 0 otherwise). Edit

Child tables:

Subclasses:

GUS Schema >> Study::Study

column	nulls?	type	description
STUDY_ID	no	NUMBER(4,0)	Edit
CONTACT_ID	no	SRes::Contact (NUMBER(12,0))	Edit
BIBLIOGRAPHIC_REFERENCE_ID		SRes::BibliographicReference (NUMBER(10,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(12,0))	Edit
SOURCE_ID		STRING(100)	Edit
NAME	no	STRING(100)	Edit
DESCRIPTION		STRING(4000)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Experimental Design
RAD::StudyAssay
PROT::StudyBioMaterial
Study::StudyDesign
Uncategorized
RAD::StudyBioMaterial

Subclasses:

webmaster@gusdb.org

GUS Schema >> PROT::StudyAssay

column	nulls?	type	description
STUDY_ASSAY_ID	no	NUMBER(8,0)	Edit
STUDY_ID	no	Study::Study (NUMBER(5,0))	Edit
ASSAY_ID	no	NUMBER(8,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

GUS Schema >> PROT::StudyBioMaterial

column	nulls?	type	description
STUDY_BIO_MATERIAL_ID	no	NUMBER(10,0)	Edit
STUDY_ID	no	Study::Study (NUMBER(10,0))	Edit
BIO_MATERIAL_ID	no	Study::BioMaterial (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> Study::StudyDesign

column	nulls?	type	description
STUDY_DESIGN_ID	no	NUMBER(5,0)	[Edit](#)
STUDY_ID	no	[Study::Study](#) (NUMBER(5,0))	[Edit](#)
NAME	no	STRING(100)	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

Experimental Design
   [PROT::StudyDesignAssay](#)
   [Study::StudyDesignDescription](#)
   [Study::StudyDesignType](#)
   [Study::StudyFactor](#)

Subclasses:

[webmaster@gusdb.org](#)

GUS Schema >> PROT::StudyDesignAssay

column	nulls?	type	description
STUDY_DESIGN_ASSAY_ID	no	NUMBER(8,0)	Edit
STUDY_DESIGN_ID	no	Study::StudyDesign (NUMBER(5,0))	Edit
ASSAY_ID	no	NUMBER(8,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> Study::StudyDesignDescription

column	nulls?	type	description
STUDY_DESIGN_DESCRIPTION_ID	no	NUMBER(5,0)	Edit
STUDY_DESIGN_ID	no	Study::StudyDesign (NUMBER(5,0))	Edit
DESCRIPTION_TYPE	no	STRING(100)	Edit
DESCRIPTION	no	STRING(4000)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> Study::StudyDesignType

column	nulls?	type	description
STUDY_DESIGN_TYPE_ID	no	NUMBER(6,0)	Edit
STUDY_DESIGN_ID	no	Study::StudyDesign (NUMBER(5,0))	Edit
ONTOLOGY_ENTRY_ID	no	Study::OntologyEntry (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> Study::StudyFactor

column	nulls?	type	description
STUDY_FACTOR_ID	no	NUMBER(5,0)	Edit
STUDY_DESIGN_ID	no	Study::StudyDesign (NUMBER(5,0))	Edit
STUDY_FACTOR_TYPE_ID	no	Study::OntologyEntry (NUMBER(12,0))	Edit
NAME	no	STRING(100)	Edit
DESCRIPTION		STRING(500)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Experimental Design
    RAD::StudyFactorValue

Subclasses:

---

webmaster@gusdb.org

gusdb.org | Schema Browser

GUS Schema >> PROT::StudyFactorValue

column	nulls?	type	description
STUDY_FACTOR_VALUE_ID	no	NUMBER(8,0)	Edit
STUDY_FACTOR_ID	no	Study::StudyFactor (NUMBER(5,0))	Edit
ASSAY_ID	no	PROT::Assay (NUMBER(12,0))	Edit
VALUE_ONTOLOGY_ENTRY_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
MEASUREMENT_UNIT_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
MEASUREMENT_TYPE		STRING(10)	Edit
MEASUREMENT_KIND		STRING(20)	Edit
STRING_VALUE		STRING(100)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

http://www.gusdb.org/SchemaBrowser/table.htm?schema=PROT&table=StudyFactorValue[2/26/13 4:20:57 PM]

GUS Schema >> RAD::AssayBioMaterial

linking table between biomaterial and assays

column	nulls?	type	description
ASSAY_BIO_MATERIAL_ID	no	NUMBER(5,0)	primary key Edit
ASSAY_ID	no	RAD::Assay (NUMBER(8,0))	fk to Assay Edit
BIO_MATERIAL_ID	no	Study::BioMaterial (NUMBER(8,0))	fk to BioSource Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::AssayLabeledExtract

linking table between Assay and LabelledExtract

column	nulls?	type	description
ASSAY_LABELED_EXTRACT_ID	no	NUMBER(8,0)	primary key Edit
ASSAY_ID	no	RAD::Assay (NUMBER(8,0))	FK to Assay Edit
LABELED_EXTRACT_ID	no	Study::BioMaterial (NUMBER(8,0))	FK to LabeledExtract Edit
CHANNEL_ID	no	Study::OntologyEntry (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> Study::BioMaterialCharacteristic

column	nulls?	type	description
BIO_MATERIAL_CHARACTERISTIC_ID	no	NUMBER(5,0)	Edit
BIO_MATERIAL_ID	no	Study::BioMaterial (NUMBER(8,0))	Edit
ONTOLOGY_ENTRY_ID	no	Study::OntologyEntry (NUMBER(10,0))	Edit
VALUE		STRING(100)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Biomaterials
Study::BMCRelationship

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> PROT::BioMaterialMeasurement

column	nulls?	type	description
BIO_MATERIAL_MEASUREMENT_ID	no	NUMBER(10,0)	Edit
TREATMENT_ID	no	NUMBER(10,0)	Edit
BIO_MATERIAL_ID	no	Study::BioMaterial (NUMBER(8,0))	Edit
VALUE		FLOAT	Edit
UNIT_TYPE_ID		NUMBER(5,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Assay
   PROT::AssayInput
   PROT::AssayProduct

Subclasses:

webmaster@gusdb.org

GUS Schema >> Study::BioSample (subclass of Study::BioMaterial)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
BIO_MATERIAL_ID	no	NUMBER(8,0)	Edit
LABEL_METHOD_ID		RAD::LabelMethod (NUMBER(4,0))	Edit
TAXON_ID		SRes::Taxon (NUMBER(10,0))	Edit
BIO_SOURCE_PROVIDER_ID		SRes::Contact (NUMBER(12,0))	Edit
BIO_MATERIAL_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
NAME		STRING(100)	Edit
DESCRIPTION		STRING(500)	Edit

Child tables:

Subclasses:

GUS Schema >> Study::BioMaterial

column	nulls?	type	description
BIO_MATERIAL_ID	no	NUMBER(8,0)	Edit
LABEL_METHOD_ID		RAD::LabelMethod (NUMBER(4,0))	Edit
TAXON_ID		SRes::Taxon (NUMBER(10,0))	Edit
BIO_SOURCE_PROVIDER_ID		SRes::Contact (NUMBER(12,0))	Edit
BIO_MATERIAL_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Experimental Design
    PROT::StudyBioMaterial
Biomaterials
    RAD::AssayBioMaterial
    RAD::AssayLabeledExtract
    Study::BioMaterialCharacteristic
    RAD::BioMaterialMeasurement
    RAD::Treatment
Uncategorized
    RAD::StudyBioMaterial

Subclasses:

Study::LabeledExtract
Study::BioSource
Study::BioSample

GUS Schema >> Study::BioSource (subclass of Study::BioMaterial)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
BIO_MATERIAL_ID	no	NUMBER(8,0)	Edit
LABEL_METHOD_ID		RAD::LabelMethod (NUMBER(4,0))	Edit
TAXON_ID		SRes::Taxon (NUMBER(10,0))	Edit
BIO_SOURCE_PROVIDER_ID		SRes::Contact (NUMBER(12,0))	Edit
BIO_MATERIAL_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
NAME		STRING(100)	Edit
DESCRIPTION		STRING(500)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

[GUS Schema](#) >> Study::BMCRelationship

column	nulls?	type	description
BMC_RELATIONSHIP_ID	no	NUMBER(5,0)	[Edit](#)
SUBJECT_BMC_ID		[Study::BioMaterialCharacteristic](#) (NUMBER(5,0))	[Edit](#)
PREDICATE_BMC_ID		[Study::BioMaterialCharacteristic](#) (NUMBER(5,0))	[Edit](#)
OBJECT_BMC_ID		[Study::BioMaterialCharacteristic](#) (NUMBER(5,0))	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> Study::LabeledExtract (subclass of Study::BioMaterial)

Light yellow rows indicate columns provided from the superclass.

column	nulls?	type	description
BIO_MATERIAL_ID	no	NUMBER(8,0)	Edit
LABEL_METHOD_ID		RAD::LabelMethod (NUMBER(4,0))	Edit
TAXON_ID		SRes::Taxon (NUMBER(10,0))	Edit
BIO_SOURCE_PROVIDER_ID		SRes::Contact (NUMBER(12,0))	Edit
BIO_MATERIAL_TYPE_ID		Study::OntologyEntry (NUMBER(10,0))	Edit
SUBCLASS_VIEW	no	STRING(27)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
NAME		STRING(100)	Edit
DESCRIPTION		STRING(500)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

[GUS Schema](#) >> RAD::Treatment

Holds an ordered set of wet-lab treatments on BioMaterials

column	nulls?	type	description
TREATMENT_ID	no	NUMBER(10,0)	primary key Edit
ORDER_NUM	no	NUMBER(3,0)	The ordering number for this treatment with respect to set of treatments Edit
BIO_MATERIAL_ID	no	Study::BioMaterial (NUMBER(8,0))	FK to the BioMaterial directly generated by this treatment or generated by a series of treaments to which this treatment belongs. Edit
TREATMENT_TYPE_ID	no	Study::OntologyEntry (NUMBER(10,0))	FK to an ontology of possible types of treatments Edit
PROTOCOL_ID		RAD::Protocol (NUMBER(10,0))	FK to a structured protocol for this treatment Edit
NAME		STRING(100)	An optional name for the treatment Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Biomaterials
   RAD::BioMaterialMeasurement
   RAD::TreatmentParam

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::TreatmentParam

Holds the instance value for a ProtocolParam of a Protocol that the owning Treatment is an instance of

column	nulls?	type	description
TREATMENT_PARAM_ID	no	NUMBER(10,0)	Primary key Edit
TREATMENT_ID	no	RAD::Treatment (NUMBER(10,0))	FK to the oenign treatment Edit
PROTOCOL_PARAM_ID	no	RAD::ProtocolParam (NUMBER(10,0))	FK to the ProtocolParam definition Edit
VALUE	no	STRING(100)	A value for the ProtocolParam Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> RAD::IntegrityStatInput

Each entry in this table describes one of the inputs to a particular integrity statistic run involving a trusted set of values in RAD and a newly added set of corresponding values, to catch possible inconsistencies with the loading of the latter. For example, if the integrity statistic computes the correlation between raw and normalized log2 ratios in a given 2-channel array experiment to check that the normalized values were properly loaded, for each element on the array there are 3 inputs: the raw value of the numerator (trusted input), the raw value of the denominator (trusted input), and the value of the normalized log2 ratio.

column	nulls?	type	description
INTEGRITY_STAT_INPUT_ID	no	NUMBER(10,0)	primary key Edit
INTEGRITY_STATISTIC_ID	no	RAD::IntegrityStatistic (NUMBER(8,0))	FK to IntegrityStatistic, pointing to the value of the statistic for which the current entry is one of the inputs. Edit
INPUT_TABLE_ID	no	Core::TableInfo (NUMBER(10,0))	FK to TableInfo, pointing to the table for which the id held in input_row_id is primary key. Edit
INPUT_ROW_ID	no	NUMBER(10,0)	The id, in the table pointed to by integrity_table_id, of the row representing the current input. Edit
ROW_DESIGNATION		STRING(200)	A designation to be used, if the current input is a trusted input, to illustrate how this input is employed in the IntegrityStatistic.trusted_input_formula. Edit
IS_TRUSTED_INPUT		NUMBER(1,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> RAD::IntegrityStatistic

Each entry in this table contains information about the statistic used and the result of a particular integrity statistic run involving a trusted set of values in RAD and a newly added set of corresponding values, to catch possible inconsistencies with the loading of the latter. An example is to compute the correlation between raw and normalized log2 ratios in a given 2-channel array experiment to check that the normalized values were properly loaded.

column	nulls?	type	description
INTEGRITY_STATISTIC_ID	no	NUMBER(8,0)	primary key Edit
STATISTIC_METHOD	no	STRING(200)	This describes which statistic was used, e.g. correlation. Edit
TRUSTED_INPUT_FORMULA		STRING(200)	This explains how the trusted inputs were combined prior to be passed to the statistical computation, in terms of their IntegrityStatInput.row_designations. For example, if the integrity statistic computes the correlation between raw and normalized log2 ratios in a given 2-channel array experiment, for each element on the array there are 3 inputs: the raw value of the numerator (trusted), say with row_designation n, the raw value of the denominator (trusted), say with raw_designation d, and the value of the normalized log2 ratio. The trusted_input_formula in this case is log2(n/d). Edit
VALUE	no	STRING(100)	The value of the resulting statistic. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Integrity
    RAD::IntegrityStatInput

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> Core::Algorithm

column	nulls?	type	description
ALGORITHM_ID	no	NUMBER(5,0)	Edit
NAME	no	STRING(100)	Edit
DESCRIPTION		STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

NA Sequence Features
[DoTS::NAFeature](#)
AA Sequence Features
[DoTS::AAFeature](#)
Central Dogma
[DoTS::ProteinProperty](#)
Algorithm
[Core::AlgorithmImplementation](#)
Uncategorized
[TESS::SequenceFeature](#)

Subclasses:

[webmaster@gusdb.org](#)

gusdb.org | Schema Browser

[GUS Schema](#) >> **Core::AlgorithmImplementation**

column	nulls?	type	description
ALGORITHM_IMPLEMENTATION_ID	no	NUMBER(5,0)	Edit
ALGORITHM_ID	no	[Core::Algorithm](#) (NUMBER(5,0))	Edit
VERSION		STRING(10)	Edit
CVS_REVISION		STRING(20)	Edit
CVS_TAG		STRING(100)	Edit
EXECUTABLE		STRING(255)	Edit
EXECUTABLE_MD5		STRING(32)	Edit
DESCRIPTION		STRING(500)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Algorithm
[Core::AlgorithmInvocation](#)
[Core::AlgorithmParamKey](#)

Subclasses:

[webmaster@gusdb.org](#)

http://www.gusdb.org/SchemaBrowser/table.htm?schema=Core&table=AlgorithmImplementation[2/26/13 4:22:35 PM]

[GUS Schema](#) >> Core::AlgorithmInvocation

column	nulls?	type	description
ALGORITHM_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)
ALGORITHM_IMPLEMENTATION_ID	no	[Core::AlgorithmImplementation](#) (NUMBER(5,0))	[Edit](#)
START_TIME	no	DATE	[Edit](#)
END_TIME	no	DATE	[Edit](#)
CPUS_USED		NUMBER(5,0)	[Edit](#)
CPU_TIME		FLOAT	[Edit](#)
RESULT	no	STRING(255)	[Edit](#)
COMMENT_STRING		STRING(255)	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

Algorithm
  [Core::AlgorithmParam](#)
Uncategorized
  [Core::AnalysisAlgorithm](#)
  [DoTS::PhylogeneticProfileSet](#)
  [DoTS::SecondaryStructure](#)

Subclasses:

webmaster@gusdb.org

GUS Schema >> Core::AlgorithmParam

column	nulls?	type	description
ALGORITHM_PARAM_ID	no	NUMBER(10,0)	Edit
ALGORITHM_PARAM_KEY_ID	no	Core::AlgorithmParamKey (NUMBER(5,0))	Edit
STRING_VALUE		STRING(1000)	Edit
FLOAT_VALUE		FLOAT	Edit
INT_VALUE		NUMBER(12,0)	Edit
BOOLEAN_VALUE		NUMBER(1,0)	Edit
TABLE_ID_VALUE		NUMBER(5,0)	Edit
ID_VALUE		NUMBER(10,0)	Edit
DATE_VALUE		DATE	Edit
IS_DEFAULT	no	NUMBER(1,0)	Edit
ORDER_NUM	no	NUMBER(5,0)	Edit
ALGORITHM_INVOCATION_ID	no	Core::AlgorithmInvocation (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> Core::AlgorithmParamKey

This table stores descriptions of the parameters for an Core::AlgorithmImplementation.

column	nulls?	type	description
ALGORITHM_PARAM_KEY_ID	no	NUMBER(5,0)	Edit
ALGORITHM_IMPLEMENTATION_ID	no	Core::AlgorithmImplementation (NUMBER(5,0))	Edit
ALGORITHM_PARAM_KEY	no	STRING(60)	Edit
ALGORITHM_PARAM_KEY_TYPE_ID	no	Core::AlgorithmParamKeyType (NUMBER(3,0))	Edit
IS_LIST_VALUED	no	NUMBER(1,0)	When set to true, the parameters takes a list of values. Edit
DESCRIPTION		STRING(1024)	Contains a short hint of what the parameter controls. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Algorithm
    Core::AlgorithmParam

Subclasses:

webmaster@gusdb.org

GUS Schema >> Core::AlgorithmParamKeyType

column	nulls?	type	description
ALGORITHM_PARAM_KEY_TYPE_ID	no	NUMBER(3,0)	Edit
TYPE	no	STRING(15)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Learning Models
   TESS::ParameterValue
Algorithm
   Core::AlgorithmParamKey

Subclasses:

webmaster@gusdb.org

GUS Schema >> SRes::DbRef

External reference entries

column	nulls?	type	description
DB_REF_ID	no	NUMBER(10,0)	Primary key column Edit
EXTERNAL_DATABASE_RELEASE_ID	no	SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
PRIMARY_IDENTIFIER		STRING(100)	Primary identification string for the entry Edit
LOWERCASE_PRIMARY_IDENTIFIER		STRING(100)	Edit
SECONDARY_IDENTIFIER		STRING(50)	The secondary identification string for the entry Edit
LOWERCASE_SECONDARY_IDENTIFIER		STRING(100)	Edit
GENE_SYMBOL		STRING(50)	Edit
CHROMOSOME		STRING(2)	Edit
CENTIMORGANS		NUMBER(22,0)	Edit
REMARK		STRING(4000)	A comment Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

NA Sequence
   DoTS::DbRefNASequence
AA Sequence
   DoTS::DbRefPfamEntry
Feature Relations
   DoTS::DbRefNAFeature
Bibliographic
   SRes::Reference
Uncategorized
   DoTS::AASequenceDbRef

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> SRes::ExternalDatabase

column	nulls?	type	description
EXTERNAL_DATABASE_ID	no	NUMBER(10,0)	Edit
NAME	no	STRING(80)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Motifs
    DoTS::RejectedMotif
External Database
    SRes::ExternalDatabaseRelease

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> SRes::ExternalDatabaseEntry

column	nulls?	type	description
EXTERNAL_DATABASE_ENTRY_ID	no	NUMBER(10,0)	Edit
EXTERNAL_DATABASE_RELEASE_ID	no	SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
EXTERNAL_PRIMARY_IDENTIFIER		STRING(255)	Edit
EXTERNAL_SECONDARY_IDENTIFIER		STRING(255)	Edit
NAME	no	STRING(40)	Edit
DESCRIPTION		STRING(255)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> SRes::ExternalDatabaseLink

column	nulls?	type	description
EXTERNAL_DATABASE_LINK_ID	no	NUMBER(10,0)	Edit
LINK_TABLE_ID	no	Core::TableInfo (NUMBER(5,0))	Edit
LINK_PRIMARY_ID	no	NUMBER(10,0)	Edit
EXTERNAL_DATABASE_RELEASE_ID	no	SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
EXTERNAL_PRIMARY_IDENTIFIER		STRING(255)	Edit
EXTERNAL_SECONDARY_IDENTIFIER		STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> SRes::ExternalDatabaseRelease

column	nulls?	type	description
EXTERNAL_DATABASE_RELEASE_ID	no	NUMBER(10,0)	Edit
EXTERNAL_DATABASE_ID	no	SRes::ExternalDatabase (NUMBER(10,0))	Edit
RELEASE_DATE		DATE	Edit
VERSION	no	STRING(255)	Edit
DOWNLOAD_URL		STRING(255)	Edit
ID_TYPE		STRING(255)	Edit
ID_URL		STRING(255)	Edit
SECONDARY_ID_TYPE		STRING(255)	Edit
SECONDARY_ID_URL		STRING(255)	Edit
DESCRIPTION		STRING(4000)	Edit
FILE_NAME		STRING(255)	Edit
FILE_MD5		STRING(32)	Edit
BLAST_FILE		STRING(255)	Edit
BLAST_FILE_MD5		STRING(32)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

NA Sequence
   DoTS::NASequence
AA Sequence
   DoTS::AASequence
   DoTS::NRDBEntry
NA Sequence Features
   DoTS::NAFeature
AA Sequence Features
   DoTS::AAFeature
GenBank Sequence Record
   DoTS::SecondaryAccs
Similarity
   DoTS::BLATAlignment
Motifs
   DoTS::Motif
   DoTS::RejectedMotif
Central Dogma
   DoTS::ProteinProperty
Mapping
   DoTS::EndSequencePairMap
   DoTS::FingerprintClone
   DoTS::FingerprintCloneMarker
   DoTS::FingerprintMap
   DoTS::MicroSatelliteMap
   DoTS::RHMap
   DoTS::RHMarker

Interaction
    DoTS::Complex
    DoTS::EffectorActionType
    DoTS::Interaction
    DoTS::InteractionType
    DoTS::Pathway
GO Association
    DoTS::GOAssociationInstance
Variation
    DoTS::Complementation
Regulatory Moieties
    TESS::Moiety
    TESS::MoietyInstance
    TESS::MoietyMember
    TESS::MoietySynonym
Regulatory Activities
    TESS::Activity
Models
    TESS::Model
High Volume Genome Annotation
    TESS::ArchipelagoOfInterest
Training Sets
    TESS::Footprint
Protocols
    RAD::Protocol
Data Transformations and Analyses
    TESS::Analysis
Platform
    RAD::ArrayDesign
    RAD::CompositeElement
    RAD::Element
Assay
    RAD::Assay
Experimental Design
    Study::Study
External Database
    SRes::DbRef
    SRes::ExternalDatabaseEntry
    SRes::ExternalDatabaseLink
Bibliographic
    SRes::BibliographicReference
    SRes::BibRefType
    SRes::Contact
Gene Ontology
    SRes::GOSynonym
    SRes::GOTerm
Enzymes Vocabulary
    SRes::ECPathway
    SRes::EnzymeClass
Repeat Types Vocabulary
    DoTS::RepeatType
Uncategorized
    Study::BioMaterial
    DoTS::ExternalDatabaseKeyword
    Study::OntologyEntry
    SRes::OntologyTerm
    DoTS::OpticalMap
    SRes::SAEL

Subclasses:

GUS Schema >> SRes::Abstract

column	nulls?	type	description
ABSTRACT_ID	no	NUMBER(10,0)	Edit
BIBLIOGRAPHIC_REFERENCE_ID	no	SRes::BibliographicReference (NUMBER(10,0))	Edit
ABSTRACT	no	CLOB	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::Attribution

column	nulls?	type	description
ATTRIBUTION_ID	no	NUMBER(10,0)	Edit
TABLE_ID	no	Core::TableInfo (NUMBER(10,0))	Edit
ROW_ID	no	NUMBER(10,0)	Edit
CONTACT_ID	no	SRes::Contact (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> SRes::Author

column	nulls?	type	description
AUTHOR_ID	no	NUMBER(10,0)	Edit
NAME	no	STRING(200)	Edit
FIRST		STRING(100)	Edit
MIDDLE_INITIALS		STRING(10)	Edit
LAST		STRING(100)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Bibliographic
 SRes::BibRefAuthor

Subclasses:

webmaster@gusdb.org

gusdb.org | Schema Browser

GUS Schema >> SRes::BibliographicReference

Each entry in this table represents a reference to an article, book, or other publication.

column	nulls?	type	description
BIBLIOGRAPHIC_REFERENCE_ID	no	NUMBER(10,0)	Primary key. Edit
BIB_REF_TYPE_ID	no	SRes::BibRefType (NUMBER(10,0))	FK to Sres.BibRefType. The id of the type of this bibliographic reference. Edit
TITLE	no	STRING(255)	The title of the reference. Edit
AUTHORS		STRING(500)	The names of the authors. Edit
PUBLICATION		STRING(255)	The name of the journal in which this reference was published. Edit
EDITOR		STRING(255)	The name of the editor. Edit
YEAR		NUMBER(5,0)	The year this reference was published. Edit
VOLUME		STRING(100)	The volume number of the journal in which this reference was published. Edit
ISSUE		STRING(100)	The issue number of the journal in which this reference was published. Edit
PAGES		STRING(100)	The page numbers of this reference in the journal. Edit
URI		STRING(250)	The URI for this reference. Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(50)	Edit
CONTACT_ID		SRes::Contact (NUMBER(12,0))	FK to Contact. The id of the person to contact regarding this reference. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Protocols
RAD::Protocol
Experimental Design
Study::Study
Bibliographic
SRes::Abstract
SRes::BibRefAuthor
SRes::SequenceReference

Subclasses:

webmaster@gusdb.org

GUS Schema >> SRes::BibRefAuthor

column	nulls?	type	description
BIB_REF_AUTHOR_ID	no	NUMBER(10,0)	Edit
BIBLIOGRAPHIC_REFERENCE_ID	no	SRes::BibliographicReference (NUMBER(10,0))	Edit
AUTHOR_ID	no	SRes::Author (NUMBER(10,0))	Edit
ORDER_NUM	no	NUMBER(5,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> SRes::BibRefType

Each entry in this table represents a type of bibliographic reference, e.g. an abstract, a book, journal article, or techical report, etc.

column	nulls?	type	description
BIB_REF_TYPE_ID	no	NUMBER(10,0)	Primary key. Edit
NAME	no	STRING(80)	The name identifying the reference type to which this entry refers. Edit
DESCRIPTION		STRING(500)	A brief description for this reference type. Edit
SOURCE		STRING(20)	The name of the source from which the definition of this reference type was taken. Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	FK to Sres.ExternalDatabaseRelease. The id of the source, if it is an external database. Edit
SOURCE_ID		STRING(255)	The id of this reference type in the original source from which its definition was taken. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Bibliographic
SRes::BibliographicReference

Subclasses:

webmaster@gusdb.org

gusdb.org | Schema Browser

GUS Schema >> SRes::Contact

Holds contact information (e.g. name, email address, phone number).

column	nulls?	type	description
CONTACT_ID	no	NUMBER(12,0)	Primary key. Edit
AFFILIATION_ID		SRes::Contact (NUMBER(12,0))	Pointer to SRes.Conact for the entry containing contact information for the affiliation of the contact. Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Foreign key to SRES.ExternalDatabaseRelease. Edit
SOURCE_ID		STRING(255)	the unique identifier of the entry in the external database Edit
NAME		STRING(255)	Complete name (first, middle initial and last name) of the Contact or the name of the Affiliation. Edit
FIRST		STRING(255)	First name of the Contact. If Affiliation, leave null. Edit
LAST		STRING(255)	Last name of the Contact. If Affiliation, leave null. Edit
CITY		STRING(255)	City used in the mailing address for the Contact or Affiliation. Edit
STATE		STRING(255)	State used in the mailing address for the Contact or Affiliation. Edit
COUNTRY		STRING(255)	Country used in the mailing address for the Contact or Affiliation. Edit
ZIP		STRING(255)	Zip code used in the mailing address for the Contact or Affiliation. Edit
ADDRESS1		STRING(255)	Street address of the Contact or Affiliation. Edit
ADDRESS2		STRING(255)	Additional street address information of the Contact or Affiliation. Edit
EMAIL		STRING(255)	Email address of the Contact or Affiliation. Edit
PHONE		STRING(255)	Phone number of the Contact or Affiliation. Edit
FAX		STRING(255)	Fax number of the Contact or Affiliation. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

NA Sequence
    DoTS::NASequence
Clones
    DoTS::CloneSet
    DoTS::EST
Data Transformations and Analyses
    RAD::Analysis
Platform
    RAD::ArrayDesign
Assay
    RAD::Assay
    RAD::Quantification
Experimental Design

[Study::Study](#)
Bibliographic
[DoTS::Attribution](#)
[SRes::BibliographicReference](#)
[SRes::Contact](#)
Enzymes Vocabulary
[DoTS::AASequenceEnzymeClass](#)
Administration
[Core::UserInfo](#)
Uncategorized
[Study::BioMaterial](#)

Subclasses:

GUS Schema >> SRes::Reference

column	nulls?	type	description
REFERENCE_ID	no	NUMBER(10,0)	Edit
DB_REF_ID		SRes::DbRef (NUMBER(10,0))	Edit
POSITION		STRING(100)	Edit
REMARK		STRING(100)	Edit
AUTHOR		STRING(1000)	Edit
TITLE		STRING(1000)	Edit
JOURNAL_OR_BOOK_NAME		STRING(1000)	Edit
JOURNAL_VOL		STRING(50)	Edit
JOURNAL_PAGE		STRING(50)	Edit
YEAR		NUMBER(5,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

GenBank Sequence Record
    DoTS::AASequenceRef
    DoTS::NASequenceRef

Subclasses:

webmaster@gusdb.org

gusdb.org | Schema Browser

GUS Schema >> SRes::SequenceReference

column	nulls?	type	description
SEQUENCE_REFERENCE_ID	no	NUMBER(10,0)	Edit
BIBLIOGRAPHIC_REFERENCE_ID	no	SRes::BibliographicReference (NUMBER(10,0))	Edit
POSITION		STRING(100)	Edit
REMARK		STRING(100)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::Comments

column	nulls?	type	description
COMMENTS_ID	no	NUMBER(10,0)	Edit
COMMENT_STRING	no	CLOB	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

[GUS Schema](#) >> **DoTS::Evidence**

Many many relation to any two tuples in the database to allow linking an evidence fact to it's target

column	nulls?	type	description
EVIDENCE_ID	no	NUMBER(10,0)	primary_key Edit
TARGET_TABLE_ID	no	Core::TableInfo (NUMBER(5,0))	table_id of the table that is the target of this evidence Edit
TARGET_ID	no	NUMBER(10,0)	row_id in the target table Edit
FACT_TABLE_ID	no	Core::TableInfo (NUMBER(5,0))	table_id of the table that is the fact (or evidence) for the target Edit
FACT_ID	no	NUMBER(10,0)	row_id in the fact table Edit
ATTRIBUTE_NAME		STRING(30)	name of the attribute in the target table to which this evidence may apply...if null then to the entire entry Edit
EVIDENCE_GROUP_ID	no	NUMBER(12,0)	groups evidence facts together if there are multiple lines of evidence associated with the same fact Edit
BEST_EVIDENCE	no	NUMBER(1,0)	if true this constitutes the best evidence fact for this target Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> SRes::ReviewStatus

column	nulls?	type	description
REVIEW_STATUS_ID	no	NUMBER(10,0)	Edit
NAME	no	STRING(255)	Edit
DESCRIPTION		STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

NA Sequence Features
  DoTS::NAFeature
AA Sequence Features
  DoTS::AAFeature
Central Dogma
  DoTS::Gene
  DoTS::GeneInstance
  DoTS::GeneSynonym
  DoTS::Protein
  DoTS::ProteinInstance
  DoTS::ProteinProperty
  DoTS::ProteinSynonym
  DoTS::RNA
  DoTS::RNAIConstruct
  DoTS::RNAInstance
  DoTS::RNAIPhenotype
Interaction
  DoTS::Complex
  DoTS::ComplexAnatomy
  DoTS::Interaction
GO Association
  DoTS::GOAssociation
  DoTS::GOAssociationInstance
  DoTS::GOAssocInstEvidCode
Variation
  DoTS::Allele
  DoTS::AlleleComplementation
  DoTS::AlleleInstance
  DoTS::AllelePhenotype
  DoTS::AllelePhenotypeClass
  DoTS::Complementation
Regulatory Moieties
  TESS::Moiety
  TESS::MoietyMember
Regulatory Activities
  TESS::Activity
  TESS::ActivityConditions
Models

TESS::Model
Data Transformations and Analyses
TESS::Analysis
TESS::ModelResult
TESS::PredictionResult
Learning Models
TESS::ParameterGroup
External Database
SRes::ExternalDatabaseEntry
Evidence
DoTS::Comments
Enzymes Vocabulary
DoTS::AASequenceEnzymeClass
Misc Applications
DoTS::AAMotifGOTermRule
Uncategorized
DoTS::AAMotifGOTermRuleSet
TESS::SequenceFeature

Subclasses:

webmaster@gusdb.org

GUS Schema >> SRes::Anatomy

Stores anatomy terms in a hierarchical manner through the use of the **parent_id**, which refers to a row in the same table that is one level closer to the root of the tree

column	nulls?	type	description
ANATOMY_ID	no	NUMBER(4,0)	Primary key Edit
NAME	no	STRING(80)	Anatomy term as determined by the **source. Most commonly an organ, tissue, or cell type. Edit**
HIER_LEVEL		NUMBER(3,0)	The number of steps from a root node of the anatomy tree. For example, "1" indicates that a row is a root node. "2" is a child of a root node. The parent of a row with **hier_level** of 2 has a **hier_level** of 1 Edit
PARENT_ID		SRes::Anatomy (NUMBER(4,0))	The primary key of a row in the same table that is one **hier_level** closer to 1. This attribute is key for the hierarchical structure of this table as it allows for distinguishing parent–child relationships between anatomy entries. The **parent_id** of a root node is 0 for **organism** Edit
SOURCE		STRING(20)	Source of the anatomy term. Sources from which terms were drawn include: GXD (Jackson Labs Mouse Gene Expression Database), GRAYS (Gray's Anatomy, 37th edtition), STEDMAN (Stedman's Medical Dictionary, 24th edition), CBIL (Computational Biology and Informatics Laboratory, UPENN), NISSANOV (Dr. Jonathan Nissanov, Drexel University) Edit
DESCRIPTION		STRING(255)	An optional description of the anatomy term. Edit
LEVEL_1		STRING(80)	Each level up to the **hier_level** of an entry contains the **name** attribute of that level of the hierarchy. For example, lymphatic vessel has a **hier_level** of 3, **level_1** = organ system, **level_2** = cardiovascular system, **level_3** = lymphatic vessel. Edit
LEVEL_2		STRING(80)	Edit
LEVEL_3		STRING(80)	Edit
LEVEL_4		STRING(80)	Edit
LEVEL_5		STRING(80)	Edit
LEVEL_6		STRING(80)	Edit
LEVEL_7		STRING(80)	Edit
LEVEL_8		STRING(80)	Edit
LEVEL_9		STRING(80)	Edit
LEVEL_10		STRING(80)	Edit
LEVEL_11		STRING(80)	Edit
LEVEL_12		STRING(80)	Edit
LEVEL_13		STRING(80)	Edit
LEVEL_14		STRING(80)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Assembly
    DoTS::AssemblyAnatomyPercent
Central Dogma
    DoTS::RNAAnatomy
Clones

[DoTS::AnatomyLibrary](#)
[DoTS::Library](#)
Interaction
[DoTS::ComplexAnatomy](#)
Regulatory Activities
[TESS::ActivityConditions](#)
Anatomy Vocabulary
[SRes::Anatomy](#)
[SRes::AnatomyLineage](#)
[DoTS::IsExpressed](#)
Phenotype Vocabulary
[SRes::Phenotype](#)
Uncategorized
[SRes::AnatomySAEL](#)

Subclasses:

GUS Schema >> SRes::AnatomyLineage

column	nulls?	type	description
ANATOMY_LINEAGE_ID	no	NUMBER(10,0)	Edit
LINEAGE_ID	no	SRes::Lineage (NUMBER(2,0))	Edit
ANATOMY_ID	no	SRes::Anatomy (NUMBER(4,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::IsExpressed

Materialized view of sequences that are expressed in some tissue type.

column	nulls?	type	description
IS_EXPRESSED_ID	no	NUMBER(10,0)	Primary Key column Edit
ANATOMY_ID	no	SRes::Anatomy (NUMBER(4,0))	FK to Anatomy CV table Edit
TABLE_ID	no	Core::TableInfo (NUMBER(5,0))	The table id that this tissue expression evidence is for (e.g Assembly, ExternalNASequence). Edit
ROW_ID	no	NUMBER(10,0)	The primary key id for the table refered by table_id Edit
IS_CONFIRMED		NUMBER(1,0)	More than one line of evidence was found that this sequence is expressed in tissue X. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> SRes::Lineage

column	nulls?	type	description
LINEAGE_ID	no	NUMBER(2,0)	Edit
NAME	no	STRING(80)	Edit
DESCRIPTION		STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Anatomy Vocabulary
   SRes::AnatomyLineage

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> SRes::GOEvidenceCode

column	nulls?	type	description
GO_EVIDENCE_CODE_ID	no	NUMBER(3,0)	Edit
NAME	no	STRING(3)	Edit
DESCRIPTION		STRING(1000)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

GO Association
DoTS::GOAssocInstEvidCode

Subclasses:

webmaster@gusdb.org

GUS Schema >> SRes::GORelationship

column	nulls?	type	description
GO_RELATIONSHIP_ID	no	NUMBER(10,0)	Edit
PARENT_TERM_ID	no	SRes::GOTerm (NUMBER(10,0))	Edit
CHILD_TERM_ID	no	SRes::GOTerm (NUMBER(10,0))	Edit
GO_RELATIONSHIP_TYPE_ID	no	SRes::GORelationshipType (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> SRes::GORelationshipType

column	nulls?	type	description
GO_RELATIONSHIP_TYPE_ID	no	NUMBER(10,0)	[Edit](#)
NAME	no	STRING(255)	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

Gene Ontology
[SRes::GORelationship](#)

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> SRes::GOSynonym

column	nulls?	type	description
GO_SYNONYM_ID	no	NUMBER(10,0)	Edit
EXTERNAL_DATABASE_RELEASE_ID	no	SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(32)	Edit
GO_TERM_ID	no	SRes::GOTerm (NUMBER(10,0))	Edit
TEXT	no	STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> SRes::GOTerm

column	nulls?	type	description
GO_TERM_ID	no	NUMBER(10,0)	Edit
GO_ID	no	STRING(32)	Edit
EXTERNAL_DATABASE_RELEASE_ID	no	SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(32)	Edit
NAME	no	STRING(255)	Edit
DEFINITION		STRING(4000)	Edit
COMMENT_STRING		STRING(4000)	Edit
MINIMUM_LEVEL	no	NUMBER	Edit
MAXIMUM_LEVEL	no	NUMBER	Edit
NUMBER_OF_LEVELS	no	NUMBER	Edit
ANCESTOR_GO_TERM_ID		SRes::GOTerm (NUMBER(10,0))	Edit
IS_OBSOLETE		NUMBER(1,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

GO Association
   DoTS::GOAssociation
Regulatory Activities
   TESS::Activity
Gene Ontology
   SRes::GORelationship
   SRes::GOSynonym
   SRes::GOTerm
Misc Applications
   DoTS::AAMotifGOTermRule

Subclasses:

GUS Schema >> DoTS::AASequenceEnzymeClass

column	nulls?	type	description
AA_SEQUENCE_ENZYME_CLASS_ID	no	NUMBER(12,0)	Edit
AA_SEQUENCE_ID	no	DoTS::AASequence (NUMBER(12,0))	Edit
ENZYME_CLASS_ID	no	SRes::EnzymeClass (NUMBER(12,0))	Edit
EVIDENCE_CODE	no	STRING(255)	Edit
EC_ASSIGNMENT_CONTACT_ID		SRes::Contact (NUMBER(12,0))	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> SRes::ECPathway

column	nulls?	type	description
EC_PATHWAY_ID	no	NUMBER(12,0)	Edit
SOURCE_ID	no	STRING(40)	Edit
EXTERNAL_DATABASE_RELEASE_ID	no	SRes::ExternalDatabaseRelease (NUMBER(12,0))	Edit
DESCRIPTION	no	STRING(400)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Enzymes Vocabulary
   SRes::ECPathwayEnzymeClass

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> SRes::ECPathwayEnzymeClass

column	nulls?	type	description
EC_PATHWAY_ENZYME_CLASS_ID	no	NUMBER(12,0)	Edit
EC_PATHWAY_ID	no	SRes::ECPathway (NUMBER(12,0))	Edit
ENZYME_CLASS_ID	no	SRes::EnzymeClass (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

GUS Schema >> SRes::EnzymeClass

column	nulls?	type	description
ENZYME_CLASS_ID	no	NUMBER(12,0)	Edit
DEPTH	no	NUMBER(1,0)	Edit
EC_NUMBER	no	STRING(16)	Edit
EC_NUMBER_1		NUMBER(3,0)	Edit
EC_NUMBER_2		NUMBER(3,0)	Edit
EC_NUMBER_3		NUMBER(3,0)	Edit
EC_NUMBER_4		NUMBER(3,0)	Edit
DESCRIPTION		STRING(128)	Edit
PARENT_ID		SRes::EnzymeClass (NUMBER(12,0))	Edit
EXTERNAL_DATABASE_RELEASE_ID	no	SRes::ExternalDatabaseRelease (NUMBER(12,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Enzymes Vocabulary
>   DoTS::AASequenceEnzymeClass
>   SRes::ECPathwayEnzymeClass
>   SRes::EnzymeClass
>   SRes::EnzymeClassAttribute

Subclasses:

webmaster@gusdb.org

GUS Schema >> SRes::EnzymeClassAttribute

column	nulls?	type	description
ENZYME_CLASS_ATTRIBUTE_ID	no	NUMBER(12,0)	Edit
ENZYME_CLASS_ID		SRes::EnzymeClass (NUMBER(12,0))	Edit
ATTRIBUTE_NAME	no	STRING(32)	Edit
ATTRIBUTE_VALUE	no	STRING(4000)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> SRes::Taxon

Holds information from the NCBI Taxonomy.

column	nulls?	type	description
TAXON_ID	no	NUMBER(10,0)	Primary key. Edit
NCBI_TAX_ID		NUMBER(10,0)	Primary taxon identifier used by NCBI. Edit
PARENT_ID		SRes::Taxon (NUMBER(10,0))	Pointer to SRes.Taxon. Edit
RANK	no	STRING(255)	Scientific classification category e.g.species, genus, kindom, etc. Edit
GENETIC_CODE_ID		SRes::GeneticCode (NUMBER(10,0))	Foreign key to SRes.GeneticCode Edit
MITOCHONDRIAL_GENETIC_CODE_ID		SRes::GeneticCode (NUMBER(10,0))	Foreign key to SRes.GeneticCode, refers to mitochondrial genome Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

NA Sequence
   DoTS::NASequence
AA Sequence
   DoTS::AASequence
   DoTS::NRDBEntry
GenBank Sequence Record
   DoTS::AASequenceTaxon
Similarity
   DoTS::BLATAlignment
Assembly
   DoTS::AssemblyAnatomyPercent
Mapping
   DoTS::EndSequencePairMap
   DoTS::MicroSatelliteMap
   DoTS::RHMap
   DoTS::RHMarker
Clones
   DoTS::Library
Variation
   DoTS::Complementation
Regulatory Moieties
   TESS::Moiety
Regulatory Activities
   TESS::ActivityConditions
Taxon Vocabulary
   SRes::Taxon
   SRes::TaxonName
Developmental Stage Vocabulary
   SRes::DevelopmentalStage
Uncategorized

[Study::BioMaterial](#)
[DoTS::OpticalMap](#)
[DoTS::PhylogeneticProfileMember](#)

Subclasses:

GUS Schema >> SRes::TaxonName

Holds information about the name of the taxon referred to by the taxon_id.

column	nulls?	type	description
TAXON_NAME_ID	no	NUMBER(10,0)	Primary key. Edit
TAXON_ID	no	SRes::Taxon (NUMBER(10,0))	Foreign key to SRes.Taxon. Edit
NAME	no	STRING(255)	text descriptor for the entry Edit
UNIQUE_NAME_VARIANT		STRING(255)	The unique variant of this name if name not unique Edit
NAME_CLASS		STRING(255)	Type of name e.g. scientific name, synonym, mispelling Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> SRes::DevelopmentalStage

Describes the developmental stage of organisms that have such stages

column	nulls?	type	description
DEVELOPMENTAL_STAGE_ID	no	NUMBER(4,0)	Primary key Edit
PARENT_ID		SRes::DevelopmentalStage (NUMBER(4,0))	Allows for the hierarchical structure of this table by indicating the primary key (**developmental_stage_id**) of the row in this same table that is one step closer to the root of the tree. Edit
TAXON_ID		SRes::Taxon (NUMBER(12,0))	Foreign key to the SRes::Taxon table indicating the organism to which the developmental stage refers. Edit
NAME		STRING(100)	The name of the developmental stage as determined by the **source**. For example, Theiler Stage 12. Edit
SOURCE		STRING(255)	Indicates the source of the developmental stage **name**. Example: "Bard et al. An internet-accessible database of mouse developmental anatomy based on a systematic nomenclature. Mechanisms of Development 74:111–120." Edit
DEFINITION		STRING(500)	Definition of the developmental stage **name** as determined by the **source**. For example, the definition of Theiler Stage 12 is: "1–4 somites, allantois extends, 1st brancial arch, heart starts to form, foregut pocket visible, preotic sulcus (at 2–3 somite stage), Days post conception (range): 8 (7.5–8.75)." Edit
LEVEL_1		STRING(50)	Each level contains the **name** attribute of that level of the hierarchy. The highest level that is entered should represent the name of the selected row. For example, trophozoite has **level_1** = blood phase and **level_2** = trophozoite. Edit
LEVEL_2		STRING(50)	Edit
LEVEL_3		STRING(50)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Clones
   DoTS::DevStageLibrary
Regulatory Activities
   TESS::ActivityConditions
Developmental Stage Vocabulary
   SRes::DevelopmentalStage
Phenotype Vocabulary
   SRes::Phenotype

Subclasses:

webmaster@gusdb.org

GUS Schema >> SRes::Disease

Describes a disease process

column	nulls?	type	description
DISEASE_ID	no	NUMBER(4,0)	Primary key Edit
PARENT_ID		SRes::Disease (NUMBER(4,0))	Allows for the hierarchical structure of this table by indicating the primary key (**disease_id**) of the row in this same table that is one step closer to the root of the tree. Edit
SOURCE		STRING(255)	Indicates the source of the disease **name** and **definition**. For example, "CDC ICD-9 (KEGG version)" Edit
NAME	no	STRING(255)	The name of the disease as determined by the **source**. For example, "Isolated tracheal or broncial tuberculosis". Edit
DEFINITION		STRING(500)	Optional definition of the disease. Edit
LEVEL_1		STRING(255)	Each level contains the **name** attribute of that level of the hierarchy. The highest level that is entered should represent the name of the selected row. For example, disease name Isolated tracheal or broncial tuberculosis has **level_1** = INFECTIOUS AND PARASITIC DISEASES and **level_2** = TUBERCULOSIS Edit
LEVEL_2		STRING(255)	Edit
LEVEL_3		STRING(255)	Edit
LEVEL_4		STRING(255)	Edit
LEVEL_5		STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Regulatory Activities
    TESS::ActivityConditions
Disease Vocabulary
    SRes::Disease

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> SRes::GeneticCode

column	nulls?	type	description
GENETIC_CODE_ID	no	NUMBER(10,0)	Edit
NCBI_GENETIC_CODE_ID	no	NUMBER(10,0)	Edit
ABBREVIATION		STRING(255)	Edit
NAME		STRING(255)	Edit
CODE	no	STRING(255)	Edit
STARTS	no	STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Taxon Vocabulary
   SRes::Taxon

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> SRes::Mutagen

column	nulls?	type	description
MUTAGEN_ID	no	NUMBER(10,0)	Edit
NAME	no	STRING(100)	Edit
DESCRIPTION		STRING(1000)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Variation
   DoTS::Allele

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> SRes::PATOAttribute

column	nulls?	type	description
PATO_ATTRIBUTE_ID	no	NUMBER(8,0)	Edit
SOURCE		STRING(255)	Edit
NAME	no	STRING(100)	Edit
DEFINITION		STRING(500)	Edit
LEVEL_1		STRING(50)	Edit
LEVEL_2		STRING(50)	Edit
LEVEL_3		STRING(50)	Edit
LEVEL_4		STRING(50)	Edit
LEVEL_5		STRING(50)	Edit
LEVEL_6		STRING(50)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Phenotype Vocabulary
   [SRes::Phenotype](#)

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> SRes::Phenotype

column	nulls?	type	description
PHENOTYPE_ID	no	NUMBER(8,0)	Edit
PARENT_ID		SRes::Phenotype (NUMBER(8,0))	Edit
PATO_ATTRIBUTE_ID		SRes::PATOAttribute (NUMBER(8,0))	Edit
ANATOMY_ID		SRes::Anatomy (NUMBER(4,0))	Edit
CELL_TYPE_ID		SRes::Anatomy (NUMBER(4,0))	Edit
DEVELOPMENTAL_STAGE_ID		SRes::DevelopmentalStage (NUMBER(4,0))	Edit
DESCRIPTION		STRING(500)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Central Dogma
    DoTS::RNAIPhenotype
Variation
    DoTS::AllelePhenotype
Regulatory Activities
    TESS::ActivityConditions
Phenotype Vocabulary
    SRes::Phenotype

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> SRes::PhenotypeClass

column	nulls?	type	description
PHENOTYPE_CLASS_ID	no	NUMBER(10,0)	[Edit](#)
CLASS	no	STRING(100)	[Edit](#)
DESCRIPTION		STRING(1000)	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

Variation
   [DoTS::AllelePhenotypeClass](#)

Subclasses:

GUS Schema >> SRes::SequenceOntology

column	nulls?	type	description
SEQUENCE_ONTOLOGY_ID	no	NUMBER(10,0)	Edit
SO_ID	no	STRING(12)	Edit
ONTOLOGY_NAME	no	STRING(30)	Edit
SO_VERSION	no	STRING(255)	Edit
SO_CVS_VERSION	no	STRING(10)	Edit
TERM_NAME	no	STRING(255)	Edit
DEFINITION	no	STRING(2000)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

NA Sequence
DoTS::NASequence
AA Sequence
DoTS::AASequence
NA Sequence Features
DoTS::NAFeature
AA Sequence Features
DoTS::AAFeature
Central Dogma
DoTS::Gene
DoTS::Protein
DoTS::RNA

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::RepeatType

column	nulls?	type	description
REPEAT_TYPE_ID	no	NUMBER(10,0)	Edit
PARENT_ID	no	DoTS::RepeatType (NUMBER(10,0))	Edit
HIERARCHY	no	NUMBER(3,0)	Edit
NAME	no	STRING(80)	Edit
DESCRIPTION	no	STRING(255)	Edit
EXTERNAL_DATABASE_RELEASE_ID		SRes::ExternalDatabaseRelease (NUMBER(10,0))	Edit
SOURCE_ID		STRING(32)	Edit
EXEMPLAR_NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(10,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

AA Sequence Features
    DoTS::AAFeature
Repeat Types Vocabulary
    DoTS::RepeatType

Subclasses:

---

webmaster@gusdb.org

[GUS Schema](#) >> DoTS::SequenceType

column	nulls?	type	description
SEQUENCE_TYPE_ID	no	NUMBER(4,0)	Edit
NUCLEOTIDE_TYPE	no	CHARACTER(7)	Edit
SUB_TYPE		STRING(20)	Edit
STRAND		CHARACTER(2)	Edit
HIERARCHY	no	NUMBER(3,0)	Edit
PARENT_SEQUENCE_TYPE_ID		[DoTS::SequenceType](#) (NUMBER(4,0))	Edit
NAME	no	STRING(80)	Edit
DESCRIPTION		STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

NA Sequence
   [DoTS::NASequence](#)
AA Sequence
   [DoTS::AASequence](#)
Sequence Types Vocabulary
   [DoTS::SequenceType](#)

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> Core::DatabaseDocumentation

Stores documentation describing the intended use of tables and attributes in the GUS schema

column	nulls?	type	description
DATABASE_DOCUMENTATION_ID	no	NUMBER(10,0)	Primary key Edit
TABLE_ID	no	Core::TableInfo (NUMBER(5,0))	Foreign key to Core.TableInfo. Edit
ATTRIBUTE_NAME		STRING(120)	Column name that is being documented. If this is null, then the documentation applies to the table. Edit
HTML_DOCUMENTATION	no	CLOB	Documentation for the "attribute_name" describing the intended use of the attribute. This documentation can be written using HTML tags. Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> Core::DatabaseInfo

column	nulls?	type	description
DATABASE_ID	no	NUMBER(5,0)	Edit
NAME		STRING(60)	Edit
DESCRIPTION	no	STRING(500)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Meta Info
   Core::TableInfo

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> Core::TableCategory

column	nulls?	type	description
TABLE_CATEGORY_ID	no	NUMBER(5,0)	[Edit](#)
CATEGORY_ID	no	[Core::Category](#) (NUMBER(5,0))	[Edit](#)
TABLE_ID	no	[Core::TableInfo](#) (NUMBER(5,0))	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> Core::TableInfo

column	nulls?	type	description
TABLE_ID	no	NUMBER(5,0)	Edit
NAME	no	STRING(30)	Edit
TABLE_TYPE	no	STRING(40)	Edit
PRIMARY_KEY_COLUMN		STRING(30)	Edit
DATABASE_ID	no	Core::DatabaseInfo (NUMBER(5,0))	Edit
IS_VERSIONED	no	NUMBER(1,0)	Edit
IS_VIEW	no	NUMBER(1,0)	Edit
VIEW_ON_TABLE_ID		Core::TableInfo (NUMBER(5,0))	Edit
SUPERCLASS_TABLE_ID		Core::TableInfo (NUMBER(5,0))	Edit
IS_UPDATABLE	no	NUMBER(1,0)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Similarity
DoTS::BLATAlignment
DoTS::Similarity
Assembly
DoTS::MergeSplit
Paralog and Family
DoTS::SequenceSequenceGroup
AA Ortholog
DoTS::BestSimilarityPair
Mapping
DoTS::EPCR
Interaction
DoTS::ComplexComponent
DoTS::RowSetMember
GO Association
DoTS::GOAssociation
Variation
DoTS::Complementation
Regulatory Moieties
TESS::MoietyInstance
Comments
TESS::Note
Data Transformations and Analyses
RAD::AnalysisResult
RAD::LogicalGroupLink
Platform
RAD::Control
Assay
RAD::Quantification
Integrity

[RAD::IntegrityStatInput](#)

External Database
[SRes::ExternalDatabaseLink](#)
Bibliographic
[DoTS::Attribution](#)
Evidence
[DoTS::Evidence](#)
Anatomy Vocabulary
[DoTS::IsExpressed](#)
Meta Info
[Core::DatabaseDocumentation](#)
[Core::TableCategory](#)
[Core::TableInfo](#)
Administration
[RAD::ProjectLink](#)
Text Search
[DoTS::IndexWordLink](#)
Uncategorized
[Core::AnalysisAlgorithm](#)
[DoTS::IndexWordSimLink](#)
[Study::OntologyEntry](#)

Subclasses:

GUS Schema >> Core::GroupInfo

column	nulls?	type	description
GROUP_ID	no	NUMBER(3,0)	Edit
NAME	no	STRING(30)	Edit
DESCRIPTION		STRING(255)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Administration
Core::UserGroup

Subclasses:

---

webmaster@gusdb.org

[GUS Schema](#) >> Core::ProjectInfo

column	nulls?	type	description
PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
NAME	no	STRING(40)	[Edit](#)
DESCRIPTION		STRING(255)	[Edit](#)
RELEASE		STRING(20)	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

Administration
  [RAD::ProjectLink](#)
  [Core::UserProject](#)

Subclasses:

[webmaster@gusdb.org](#)

GUS Schema >> RAD::ProjectLink

column	nulls?	type	description
PROJECT_LINK_ID	no	NUMBER(10,0)	Edit
PROJECT_ID	no	Core::ProjectInfo (NUMBER(4,0))	Edit
TABLE_ID	no	Core::TableInfo (NUMBER(5,0))	Edit
ROW_ID	no	NUMBER(10,0)	Edit
CURRENT_VERSION		STRING(4)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> Core::UserGroup

column	nulls?	type	description
USER_GROUP_ID	no	NUMBER(12,0)	Edit
USER_ID	no	Core::UserInfo (NUMBER(12,0))	Edit
GROUP_ID	no	Core::GroupInfo (NUMBER(3,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

[GUS Schema](#) >> Core::UserInfo

column	nulls?	type	description
USER_ID	no	NUMBER(12,0)	[Edit](#)
LOGIN	no	STRING(30)	[Edit](#)
PASSWORD	no	STRING(50)	[Edit](#)
FIRST_NAME	no	STRING(30)	[Edit](#)
LAST_NAME	no	STRING(30)	[Edit](#)
E_MAIL	no	STRING(60)	[Edit](#)
CONTACT_ID		[SRes::Contact](#) (NUMBER(12,0))	[Edit](#)
MODIFICATION_DATE	no	DATE	[Edit](#)
USER_READ	no	NUMBER(1,0)	[Edit](#)
USER_WRITE	no	NUMBER(1,0)	[Edit](#)
GROUP_READ	no	NUMBER(1,0)	[Edit](#)
GROUP_WRITE	no	NUMBER(1,0)	[Edit](#)
OTHER_READ	no	NUMBER(1,0)	[Edit](#)
OTHER_WRITE	no	NUMBER(1,0)	[Edit](#)
ROW_USER_ID	no	NUMBER(12,0)	[Edit](#)
ROW_GROUP_ID	no	NUMBER(4,0)	[Edit](#)
ROW_PROJECT_ID	no	NUMBER(4,0)	[Edit](#)
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	[Edit](#)

Child tables:

Administration
[Core::UserGroup](#)
[Core::UserProject](#)
Misc Applications
[DoTS::AAMotifGOTermRule](#)
Uncategorized
[DoTS::AAMotifGOTermRuleSet](#)

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> Core::UserProject

column	nulls?	type	description
USER_PROJECT_ID	no	NUMBER(12,0)	Edit
USER_ID	no	Core::UserInfo (NUMBER(12,0))	Edit
PROJECT_ID	no	Core::ProjectInfo (NUMBER(4,0))	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::AAMotifGOTermRule

column	nulls?	type	description
AA_MOTIF_GO_TERM_RULE_ID	no	NUMBER(10,0)	Edit
AA_MOTIF_GO_TERM_RULE_SET_ID	no	DoTS::AAMotifGOTermRuleSet (NUMBER(10,0))	Edit
GO_TERM_ID	no	SRes::GOTerm (NUMBER(10,0))	Edit
NUMBER_OF_ANNOTATED_PROTEINS	no	NUMBER(12,0)	Edit
DEFINING	no	NUMBER(1,0)	Edit
REVIEW_STATUS_ID		SRes::ReviewStatus (NUMBER(12,0))	Edit
REVIEWER_ID		Core::UserInfo (NUMBER(12,0))	Edit
CONFIDENCE		STRING(10)	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Schema >> DoTS::GeneFeatureSeqOverlap

column	nulls?	type	description
GENE_Feature_SEQ_OVERLAP_ID	no	NUMBER(12,0)	Edit
GENE_NA_Feature_ID	no	DoTS::NAFeature (NUMBER(12,0))	Edit
GENOMIC_NA_SEQUENCE_ID	no	DoTS::NASequence (NUMBER(12,0))	Edit
SEQ_NA_SEQUENCE_ID		DoTS::NASequence (NUMBER(12,0))	Edit
END_SEQUENCE_PAIR_MAP_ID		DoTS::EndSequencePairMap (NUMBER(12,0))	Edit
EPCR_ID		DoTS::EPCR (NUMBER(12,0))	Edit
SIMILARITY_ID		DoTS::Similarity (NUMBER(10,0))	Edit
BLAT_ALIGNMENT_ID		DoTS::BLATAlignment (NUMBER(12,0))	Edit
GENE_OVERLAP	no	NUMBER(12,0)	Edit
GENE_CODING_OVERLAP	no	NUMBER(12,0)	Edit
GENE_LENGTH	no	NUMBER(12,0)	Edit
GENE_CODING_LENGTH	no	NUMBER(12,0)	Edit
SEQUENCE_LENGTH	no	NUMBER(12,0)	Edit
GENE_NUM_EXONS	no	NUMBER(12,0)	Edit
GENE_NUM_EXONS_HIT	no	NUMBER(12,0)	Edit
GENE_PERCENT_COVERED	no	FLOAT	Edit
GENE_PERCENT_CODING_COVERED	no	FLOAT	Edit
SEQUENCE_PERCENT_COVERED	no	FLOAT	Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::IndexWord

Table of words that have been indexed

column	nulls?	type	description
INDEX_WORD_ID	no	NUMBER(10,0)	primary_key Edit
WORD	no	STRING(255)	the word that has been indexed Edit
DESCRIPTION		STRING(255)	a description of the word (in most cases will be null) Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Text Search
  DoTS::IndexWordLink
Uncategorized
  DoTS::IndexWordSimLink

Subclasses:

---

webmaster@gusdb.org

GUS Schema >> DoTS::IndexWordLink

Relation between an IndexWord and the tuple that has been indexed

column	nulls?	type	description
INDEX_WORD_LINK_ID	no	NUMBER(10,0)	primary_key Edit
INDEX_WORD_ID	no	DoTS::IndexWord (NUMBER(10,0))	foreign key to IndexWord Edit
TARGET_TABLE_ID	no	Core::TableInfo (NUMBER(5,0))	table_id of the table that this relation joins Edit
TARGET_ID	no	NUMBER(10,0)	row_id in the target table Edit
MODIFICATION_DATE	no	DATE	Edit
USER_READ	no	NUMBER(1,0)	Edit
USER_WRITE	no	NUMBER(1,0)	Edit
GROUP_READ	no	NUMBER(1,0)	Edit
GROUP_WRITE	no	NUMBER(1,0)	Edit
OTHER_READ	no	NUMBER(1,0)	Edit
OTHER_WRITE	no	NUMBER(1,0)	Edit
ROW_USER_ID	no	NUMBER(12,0)	Edit
ROW_GROUP_ID	no	NUMBER(4,0)	Edit
ROW_PROJECT_ID	no	NUMBER(4,0)	Edit
ROW_ALG_INVOCATION_ID	no	NUMBER(12,0)	Edit

Child tables:

Subclasses:

webmaster@gusdb.org

GUS Developer's Guide

GUS Developer's Guide

The Genomics Unified Schema and Application Framework are subject to various license terms and copyrights as outlined in the LICENSE file provided with the software.

Table of Contents

List of Tables

List of Examples

Chapter 1. Developing GUS Plugins

Steve Fischer `<sfischer@pcbi.upenn.edu>`

GUS Plugins

GUS plugins are Perl programs that load data into GUS. They are written using the Plugin API (the section called "The Plugin API"). You may use plugins that are bundled with the GUS distribution or you may write your own.

The standard GUS practice is to use only plugins, not straight SQL or bulk loading, to load the database. The reason is that plugins:

- track the data that is loaded

- copy any updated or deleted rows to "version" tables that store a history of the changes

- are known programs that can be scrutinized and used again

- have a standard documentation process so that they are easily understood

- use the Plugin API and so are easier to write than regular scripts.

Supported versus Community Plugins

The distribution of GUS comes with two types of plugins:

- *Supported* plugins:

    - are confirmed to work

    - are portable

    - are useful to sites other than the site that developed the plugin

    - meet the Plugin Standard described below

- *Community* plugins:

    - are contributed by the staff at CBIL and any other plugin developers

    - have not been reviewed with respect to the criteria for being supported

When you begin writing your plugin, use as a guideline or as a template an existing supported plugin. They are found in `$PROJECT_HOME/GUS/Supported/plugin/perl`.

The Plugin API

`Plugin.pm`: The Plugin Superclass

GUS plugins are subclasses of `GUS::PluginMgr::Plugin`. The public subroutines in `Plugin.pm` (private ones begin with an underscore) constitute the Plugin API. GUS also provides Perl objects for each table and view in the GUS schema. These are also part of the API. (the section called "The Plugin API")

The plugin's package and @ISA statements

All plugins must declare their package, using Perl's `package` statement. The package name of a plugin is derived as follows:

ProjectName::*ComponentName*::Plugin::*PluginName*

Plugins must also declare that they are subclasses of `Plugin.pm`, using Perl's `@ISA` array. The first lines of a plugin will look like this:

```
package GUS::Supported::Plugin::SubmitRow
```

```
@ISA = qw(GUS::PluginMgr::Plugin)
```

Plugin Initialization

Plugins are objects and so must have a constructor. This constructor is the `new()` method. The `new()` method has exactly two tasks to accomplish: constructing the object (and returning it), and initializing it. Construction of the object follows standard Perl practice. Initialization is handled by the `Plugin.pm` superclass method `initialize()`. the section called "The Plugin API" for details about that method.

Example 1.1. A Sample `new()` method

```
sub new {
    my ($class) = @_;
    my $self = {};

    bless($self,$class);

    $self->initialize({
        requiredDbVersion => 3.5,
        cvsRevision => '$Revision: 3009 $',
        name => ref($self),
        argsDeclaration => $argsDeclaration,
        documentation => $documentation
    });

    return $self;
}
```

The `$Revision: 3009 $` string is CVS or Subversion keyword. When the plugin is checked into source control, the repository substitutes the file's revision into that keyword. The keywords must be in single quotes to prevent Perl from interpreting `$Revision: 3009 $` as a variable.

Keeping your Plugin Current as GUS Changes

If you follow the pattern used by supported plugins, you will only ever need to change one line in the `new()` method. As you can probably tell, `initialize()` takes one argument, a reference to a hash that contains a set of parameter values. The one you will need to change is `requiredDbVersion`. As

the GUS schema evolves, you will need to review your plugin to make sure it is compatible with the latest version of GUS, upgrading it if not. When it is compatible with the new version of GUS, update `requiredDbVersion` to that version of GUS.

Declaring the plugin's command line arguments

In the example above (Example 1.1, "A Sample `new()` method"), the line

```
argsDeclaration => $argsDeclaration,
```

provides to the `initialization()` method a reference to an array, `$argsDeclaration`, that declares what command line arguments the plugin will offer. When you look at a supported plugin you will see the `$argsDeclaration` variable being set like this:

Example 1.2. Defining Command Line Arguments

```
my $argsDeclaration = [
    tableNameArg({name  => 'tablename',
                  descr => 'Table to submit to, eg, Core::UserInfo',
                  reqd  => 1,
                  constraintFunc=> undef,
                  isList =>0,
    }),

    stringArg({name  => 'attrlist',
              descr => 'List of attributes to update (comma delimited)',
              reqd  => 1,
              constraintFunc => undef,
              isList = >1,
    }),

    enumArg({name  => 'type',
            descr => 'Dimension of attributes (comma delimited)',
            reqd  => 1,
            constraintFunc => undef,
            enum => "one, two, three",
            isList => 1,
    }),

    fileArg({name  => 'matrixFile',
            descr => 'File containing weight matrix',
            reqd => 1,
            constraintFunc=> \&checkFileFormat,
            mustExist=>0,
            isList=>0,
    }),
];
```

If you look carefully at the list above you will notice that each element of it is a call to a method such as `stringArg()`. These are methods of `Plugin.pm` and they all return subclasses of `GUS::PluginMgr::Args::Arg`. In the case of `stringArg()`, it returns `GUS::PluginMgr::Args::StringArg`. All you really need to know is that there are a set of methods available for you to use when declaring your command line arguments. That is, the `argsDeclaration` parameter of the `initialize()` method expects a list of `Arg` objects. You can learn about them in detail in the Plugin API (the section called "The Plugin API")

The `Arg` objects are very powerful. They parse the command line, validate the input, handle list values,

deal with optional arguments and default values and provide for documentation of the arguments. There are two ways the `Arg` objects validate the input. First, it applies its standard validation. For example, a `FileArg` confirms that the input is a file, and throws an error otherwise. Second, if you provide a `constraintFunc`, it will run that as well, throwing an error if the plugin value violates the constraints.

Declaring the Plugin's Documentation

In a way that parallels the declaration of command line arguments, the initialize method also expects a reference to a hash that provides standardized fields that document the plugin: (Example 1.1, "A Sample `new()` method")

```
documentation => $documentation,
```

Here is a code snippet that demonstrates the standard way `$documentation` is set:

Example 1.3. Defining Plugin Documentation

```
my $purposeBrief = <<PURPOSE_BRIEF;
Load blast results from a condensed file format into the DoTS.Similarity table.
PURPOSE_BRIEF

my $purpose = <<PLUGIN_PURPOSE;
Load a set of BLAST similarities from a file in the form generated by the blastSim
PLUGIN_PURPOSE

my $tablesAffected =
    [ ['DoTS::Similarity', 'One row per similarity to a subject'],
      ['DoTS::SimilaritySpan', 'One row per similarity span (HSP)'],
    ];

my $tablesDependedOn =
    [
    ];

my $howToRestart = <<PLUGIN_RESTART;
Use the restartAlgInvs argument to provide a list of algorithm_invocation_ids that
previous runs of loading these similarities. The algorithm_invocation_id of a run
plugin is logged to stderr. If you don't have that information for a previous run
you will have to poke around in the Core.AlgorithmInvocation table and others to f
runs and their algorithm_invocation_ids.
PLUGIN_RESTART

my $failureCases = <<PLUGIN_FAILURE_CASES;
PLUGIN_FAILURE_CASES

my $notes = <<PLUGIN_NOTES;
The definition lines of the sequences involved in the BLAST (both query and subjec
begin with the na_sequence_ids of those sequences. The standard way to achieve tha
first load the sequences into GUS, using the InsertFastaSequences plugin, and then
extract them into a file with the dumpSequencesFromTable.pl command. That command
the na_sequence_id of the sequence as the first thing in the definition line.
PLUGIN_NOTES

my $documentation = { purpose=>$purpose,
                      purposeBrief=>$purposeBrief,
                      tablesAffected=>$tablesAffected,
                      tablesDependedOn=>$tablesDependedOn,
                      howToRestart=>$howToRestart,
                      failureCases=>$failureCases,
```

```
                    notes=>$notes
                 };
```

When you look at this example, you will see that a bunch of variables, such as $purposeBrief and $tablesAffected, are being set. They are used as values of the hash called $documentation. $documentation is in turn passed as a value to the initialize() method. You will also notice that Perl's HEREDOC syntax is used. The setting of the variables begins with, eg, <<PLUGIN_PURPOSE and ends with, eg, PLUGIN_PURPOSE. This is Perl's way of allowing you to create paragraph-style strings without worrying about quoting or metacharacters such as \n.

The documentation is shown to the user when he or she uses the help flag, or when he or she makes a command line error.

The documentation is formatted using Perl's documentation generation facility, pod. This means that you can include simple pod directives in your documentation to, say, emphasize a word. Run the command perldoc perlpod for more information

The `run()` Method

Plugins are run by a command called ga (which stands for "GUS application"). ga constructs the plugin (by calling its new() method) and then runs the plugin by calling its run() method.

The purpose of the run() method is to provide at a glance the structure of the plugin. It should be very concise and under no circumstances be longer than one screen. A good practice, when reasonable, is for the run() method to call high level methods that return the objects to be submitted to the database, and then to submit them in the run() method. This way, a reader of the run() method will know just what is being written to the database, which is the main purpose of a plugin.

The run() method is expected to return a string describing the result of running the plugin. An example would be "inserted 3432 sequences".

The Pointer Cache

The pointer cache is a somewhat infamous component of the GUS object layer. It is a memory management facility that was designed to steer around poor garbage collection in Perl (in 2000). Whether or not is still needed is another matter because it is part of the object layer for now. The pointer cache is a way for the plugin to re-use objects that have been allocated but are no longer in active use. Because Perl was not properly garbage collecting objects when they were no longer referred to, the memory footprint of plugins was getting huge.

As a plugin developer what you need to know is that at points in your code where you no longer need any of the GUS objects that you have created (typically at the bottom of your outermost loop, you should call the Plugin.pm method undefPointerCache(). This method clears out the cache.

If the default capacity (10000) is not enough to hold all the objects you are creating in one cycle through your logic, you can augment its size with the Plugin.pm method setPointerCacheSize().

The Plugin Standard

Portability

A supported plugin must be useful to sites other than the site that developed it. It also must run at other sites without modification.

Plugin Naming

- Plugin names begin with one of four verbs:

    - *insert* if the plugin inserts only

    - *delete* if the plugin deletes only

    - *update* if the plugin updates only

    - *load* if the plugin does any two or more of insert, delete or update

- Plugin names are concise

    - for example, a plugin named InsertNewSequences is not concise because Insert and New are re-dundant

- Plugin names are precise

    - for example, a plugin named InsertData is way too general. The name should reflect the type of data inserted

    - if a Plugin expects exactly one file type, that file type should be in the name. For example, In-sertFastaSequences.

- Plugin names are accurate

    - for example, a plugin named InsertExternalSequences is inaccurate if it can also insert internally generated sequences. A better name would be InsertSequences.

GUS Primary Keys

Plugins never directly use (hard-code) GUS primary keys, either in the body of the code or for command line argument values. Instead they use semantically meaningful alternate keys. The reason that plugins cannot use primary keys in their code is that doing so makes the plugin site specific, not portable. The reason they cannot use primary keys as values in their command line arguments is that plugins are often incorporated as steps in a pipeline (using the GUS Pipeline API described elsewhere). The pipelines should be semantically transparent so that people both on site and externally who look at the pipeline will understand it.

Application Specific Tables

Some sites augment GUS with their own application specific tables. These are not permitted in suppor-ted plugins.

Command Line Arguments

- The name of the argument should be concise and precise

- The Plugin API provides a means for you to declare arguments of different types, such integers, strings and files (the section called "Declaring the plugin's command line arguments"). Use the most appropriate type. For example, don't use a string for a file argument.

- Use camel caps (eg matrixFile) not underscores (eg matrix_file) in the names of the arguments.

Documentation

The Plugin API provides a means for you to document the plugin and its arguments. Be thorough in your documentation. the section called "Declaring the Plugin's Documentation"

Use of GUS Objects

The GUS object layer assists in writing clean plugin code. The guidelines for their use are:

- When writing data to the database, use GUS objects when possible. Avoid using SQL directly.

- When forming a relationship between two objects, use the `setParent()` or `setChildren()` method. Do not explicitly set the foreign keys of the objects.

Database Access

The GUS objects are good at writing data to the database. That is because they allow you to build up a tree structure of objects and then to simply submit the root. However they are not as useful at reading the database. You can only read one object at a time (more on this in the Guide to GUS Objects). For this reason, you will need to use SQL to efficiently read data from the database as needed by your plugin.

This is how a typical database access looks:

Example 1.4. Typical Database Access

```
my $sql =
   "SELECT $self->{primaryKeyColumn}, $self->{termColumn}
    FROM $self->{table}";

my $queryHandle = $self->getQueryHandle();
my $statementHandle = $queryHandle->prepareAndExecute($sql);

my %vocabFromDb;

while (my ($primaryKey, $term) = $sth->fetchrow_array()) {
    $vocabFromDb{$term} = $primaryKey;
}
```

The SQL is formatted on multiple lines for clarity (Perl allows this), and the SQL keywords are upper case. The Plugin API provides a method to easily get a query handle, returning a `GUS::ObjRelP::DbiDbHandle`. That object provides an easy-to-use method that prepares and executes the SQL.

Logging

The Plugin API offers a set of logging methods. They print to standard error. Use these and no other means of writing out logging messages.

Standard Output

Do not write to standard output. If your plugin generates data (such as a list of IDs already loaded, for restart) write it to a file.

Commenting

Less is more with commenting. Comment only the non-obvious. For example, do not comment a method called `getSize()` with a comment `# gets the size`. Most methods should need no commenting, as they should be self-explanatory. In many cases, if you find that you need to comment because something non-obvious needs explaining, that is a red flag indicating that your code might need simplification.

Handling Errors

There is only one permissible way to handle errors: call `die()`. Never log errors or write them to standard error or standard out. Doing that masks the error (the logs are not read reliably) so that what is really happening is the plugin is failing silently. Causing the plugin to die forces the user of the plugin or its developer to fix the problem.

When you call die, give it an informative message, including the values of the suspicious variables. Surround the variables in single quotes so that white space errors will be apparent. Provide enough information so that the user can track down the source of the problem in the input files.

If you would like your program to continue past errors, then dedicate a file or directory which will house describing the errors. The user will know that he or she must look there for a list of inputs that caused problems. Typically you use this strategy if you expect the input to be huge, and don't want to abort it because of a few errors. You may want to include as a command line argument the number of errors a user will tolerate before giving up and just aborting.

Failure Recovery and Restart

Plugins abort. They do so for many reasons. When they do, the user must be able to recover from the failure, one way or another.

A few strategies you could adopt are:

- If the plugin is inserting data (rather than inserting and updating) the plugin can check if an object that is about to be written to the database is already there. If so, it can skip that object. Because this checking will slow the plugin down, the plugin should offer a `restart` flag on the command line that turns that check on.

- If the plugin is updating it can include a command line argument that takes a list of `row_alg_invocation_ids`, one per each run of the plugin with this dataset. (Each table in GUS has a `row_alg_invocation_id` column to store the identifier of the particular run of a plugin that put data there. This is part of the automatic tracking that plugins do.) The plugin can take the same approach as the previous strategy, but, must additionally check that the object has one of the provided `row_alg_invocation_id`s.

- The plugin can store in dedicated file the identifiers of the objects it has already loaded. In this case, the plugin should offer a command line argument to ask for the name of the file.

Opening Files

A very common error is to open files without dying if the open fails. The proper way to open a file is like this:

Example 1.5. Properly Opening a File

```
open(FILE, $myFile) || die "could not open file '$myFile'\n");
```

Caching to Minimize Database Access

One of the most time consuming operations in a plugin is accessing the database. The typical flow of a plugin is that it reads the input and as it goes it constructs and submits GUS objects to the database. Some plugins additionally need to read data from the database to do their work. While it is often impossible to avoid writing to the database with each new input value, it is often possible to avoid reading it.

If most of the values of a table (or tables) will be needed then the plugin should read the table (or tables) outside the loop that processes the input. It should store the values in a hash keyed on a primary or alternate key. Storing multiple megabytes of data this way in memory should not be a problem. Gigabytes may well be a problem.

If only a few values from the table will be needed then an alternative caching strategy may be appropriate. Wrap the access to the values in a getter method, such as `getGeneType()`. This method stores values it gets in a hash. When the method is called, it first looks in the hash for the value. If the hash does not have it, then the method reads the database and stores the value in the hash to optimize future accesses.

Regular Expressions

Complicated regular expressions should be accompanied by a comment line that shows what the input string looks like. It is otherwise often very difficult to figure out what the regular expression is doing. Long regular expressions should be split into multiple lines with embedded whitespace and comments using the `/x` modifier. See the "Readability" section of Maintaining Regular Expressions [http://www.perl.com/pub/a/2004/01/16/regexps.html]

Variable and Method Names

Choosing good names for your variables and methods makes your code much more understandable. To make your code clear:

- Variable and method names should start with a lower case letter.

- Use "camel caps" (`$sequenceLength`) for variable names and method names, not underscores (`$sequence_length`).

- Variable names should be named after the type of data they hold (unless there are more than one variable for a given type, in which case they are qualified). For example a good name for a sequence would be `$sequence`

- In plugins, there are typically:

  - strings parsed from the input

  - objects created from the input (if you are using an object based parser such as Bioperl)

  - GUS object layer objects

- Input objects or strings should be named with 'input' as a prefix. For example: `$inputSequence`

- Object layer objects are named for their type, for example `$NASequence`

- Method names should be self-explanatory. A bad method name would be `process()` (what is being processed?). Don't "save keystrokes" with short names. If being self-explanatory requires using a long name, then use a long name.

Methods

Use "structured programming" when you create your methods:

- No method should ever be longer than one screen. If it is, refactor part of into its own method.

- Never repeat code. Repeated code must be in a method.

Some methods in the API are marked as deprecated. Do not use them. They are for backward compatibility only.

Syntax

- Use C and Java like syntax. Do not use weird Perl specific syntax.

- Indenting must be spaces not tabs. Two or four spaces are acceptable

- Use `$self` to refer to the object itself

- Declare method arguments using this syntax:

  `my ($self, $sequence, $length) = @_;.`

  Do not use `shift`

Application Specific Controlled Vocabularies

A controlled vocabulary (CV) is a restricted set of terms that are allowed values for a data type. They may be simple lists or they may be complex trees, graphs or ontologies. In GUS the CVs fall into two categories: standard CVs such as the Gene Ontology, and small application specific CVs such as ReviewStatus.

The complete list of application specific CVs in the GUS 3.5 schema is:

- DoTS.BlatAlignmentQuality

- DoTS.GOAssociationInstanceLOE

- DoTS.GeneInstanceCategory

- DoTS.InteractionType

- DoTS.MotifRejectionReason

- DoTS.ProteinCategory

- DoTS.ProteinInstanceCategory

- DoTS.ProteinProteinCategory

- DoTS.ProteinPropertyType

- DoTS.RNACategory

- DoTS.RNAInstanceCategory

- DoTS.RNARNACategory

- DoTS.RepeatType

- SRes.BibRefType

- SRes.ReviewStatus

Acquiring a standard CV typically involves downloading files from the CV provider and running a plugin to load it.

Application specific CVs are handled by the plugin that will use the CV. For example, a plugin that inserts bibliographic references will use the SRes.BibRefType CV. It is these plugins that are responsible for making sure that the CV they want to use is in the database.

Plugins that use CVs fall into two categories:

1. those that hard code the CV

2. those that do not hard code the CV, but, rather, get it from the input

In case 1, the plugin hard codes the CV in the Perl code.

In case 2, the plugin hard codes only a default. It also offers an optional command line argument that takes a file that contains the CV. If the user of the plugin determines that the input has an different CV than the default, the user will provide such a file.

In both cases, the plugin reads the table in GUS that contains the CV and compares it to the CV it expects to use. If the expected vocab is not found, the plugin updates the table.

Assigning an External Database Release Id

GUS is a data warehouse so it is very common for plugins to load into GUS data from another source. Whether the source is external or in-house, tracking its origin is often required. The tables in GUS that handle this are SRes.ExternalDatabase and SRes.ExternalDatabaseRelease. The former describes the database, eg, PFam, and the latter describes the particular release of the database that is being loaded, eg, 1.0.0. The data loaded will have a foreign key to the database release, which in turn has a foreign key to the database.

In order to create that relationship, the plugin must know the primary key of the external database release. To accomplish this, the plugin takes as command line arguments the name of the database and its release. It does not take the primary key of the external database release (that violates the plugin standard). The plugin passes that information to the API subroutine `getExtDbRlsId($dbName, $dbVersion)`.

If the plugin is inserting the dataset as opposed to updating it, create new entries for the database and the release by using the plugins `GUS::Supported::Plugin::InsertExternalDatabase` and `GUS::Supported::Plugin::InsertExternalDatabaseRls`.

Chapter 2. Extending the Schema

Michael Saffitz `<msaffitz@pcbi.upenn.edu>`

The GUS Schema may be extended by adding new columns to existing tables, or adding new tables and views. For the time being, adding new Schemata to GUS is not supported.

Important

Extensions to the GUS may interfere with your ability to upgrade to future releases of GUS.

Creating New Objects in the Database

The first step to extending the Schema is to create the objects within the database.

Adding New Columns to Existing Tables

When adding new columns to existing tables, it is important to maintain the existing order of the columns, and only add new columns between the existing columns and the housekeeping columns (eg before the `modification_date` column). For this reason, it will likely be necessary to rename the existing table; create the modified table; and then migrate the date from the existing table to the newly created table. As you perform this process, you should ensure that all constraints (including both "incoming" and "outgoing" foreign key constraints and primary key constraints) and that indexes on the original table are created and applied on the newly created table.

Adding New Tables

When creating new tables, it is important to include all housekeeping columns at the "end" of the definition in the proper order. All new tables should have a corresponding sequence created, with the naming convention of: *TableSchemaName*.*TableName*_SQ . All new tables must have a single column primary key constraint defined.

Adding New Views

Only views created as "subclass" views against an implementation table are supported. When creating new views, it is important to include all superclass columns in the view definition, including the housekeeping columns. Proper column ordering should be observed in the views.

Updating GUS Version Objects

If you've changed an existing GUS table, or wish to have GUS audit changes to your new tables and views, you must make the corresponding changes and/or additions to the version ("ver") tables and views.

Updating `Core.TableInfo`

GUS stores metadata for all tables and views in the `Core.TableInfo` table. Whenever you create a new table or view, you must add a corresponding row in this table. The column descriptions are:

Table 2.1. `Core.TableInfo` Description

Column Name	Description
`table_id`	The ID of this row, provided by the com `Core.TableInfo_SQ` sequence.
`name`	The name of the table. The case used here will be used at object-generation time.
`table_type`	`Standard` or `Version`, depending on whether this is a normal table or a version table
`primary_key_column`	The name of the primary key column
`database_id`	The id of the schema, found in `Core.DatabaseInfo`
`is_versioned`	1 if the table has a corresponding version table. 0 otherwise.
`is_view`	1 if the "table" is a subclass view. 0 otherwise.
`view_on_table_id`	If this "table" is a subclass view, the table id of the implementation table that this view is against.
`superclass_table_id`	If this "table" is a subclass view, the table id of the superclass view.
`is_updatable`	1 if the table is read/write, 0 if it is read only.

Rebuilding Objects

After you have completed the steps above, you must rebuild your objects. First, use the command below to signal that your table definitions have changed:

```
$ touch $PROJECT_HOME/Schema/gus_schema.xml
```

Then, reinstall GUS:

```
$ build GUS install -append
```

ReFlow User's Manual

Under construction. ReFlow is in production at the EuPathDB project and is being prepared for use externally. This document is a work-in-progress. If you have questions please drop a line to the ReFlow Users mailing list.

Overview

About this document

This document describes the ReFlow workflow system. ReFlow was first developed by the EuPathDB project as an in-house workflow system. However, it may be well suited to use by others. In addition to general ReFlow documentation, some particulars of EuPathDB's use of ReFlow are documented here. They are clearly marked, serve as primary documentation for in-house users, and as hints for external users.

Basic Concepts

ReFlow is a simple workflow system based on a dependency graph. It runs on UNIX only, and has been tested on Linux. Its primary user interface is textual (command line and log files). While it does not have a GUI, its textual tools do a good job managing very large workflows.

In overview:

- A ReFlow workflow is a directed acyclic graph
    - The nodes are *Steps*
    - The edges are dependencies. A step is run when all the steps it depends are done.

- Steps
    - execute an atom of work, such as:
        - creating a shared directory
        - acquiring data from the internet
        - loading data into a database
        - running a local analysis program
        - starting a job on a compute cluster
    - are an instance of a *step class*.
    - declare dependencies on other steps

    - are configured with parameter values

- Step classes

    - perform a particular type of work
    - are written in Perl, and are a subclass of ReFlow::Controller::WorkflowStepInvoker
    - are parameterized. When an instance of a step class is invoked by the controller, it receives a package of key-value pairs, much like a subroutine is given argument values.
    - can do any kind of work that is compatible with the WorkflowStepInvoker API. In general, this is launching one or a series of processes on the controller machine that will give a correct exit status code on completion.

- A Graph
    - is specified in a graph XML file
        - the schema is defined in RNG
        - HERE is an example
    - calls one or more Steps in the order specified by their dependencies
    - declares input parameters
    - declares constants
    - defines a local scope in which steps can
        - depend on other steps
        - have access to graph input parameters
        - have access to graph constants
    - specifies parameter values to be passed to each step
    - may call subgraphs
        - these are similar to Steps but invoke a nested graph instead of a step class instance
        - the nesting of subgraphs forms a containment hierarchy that is orthogonal to the dependency graph.
    - may be called as a root or subgraph
    - may be called multiple times within the full graph
    - may be thought of as a subroutine

- The Root Graph
    - is the top of the nesting hierarchy.
    - is no different than any other graph, except that it
        - is the top of the nesting
        - is passed its input parameter values from the `rootParams.prop` file instead of a calling graph

- is allowed to contain a call to the special subgraph `<globalSubgraph>` (see XXXX below).

- The Full Graph
  - is the dependency graph obtained by expanding all the nested subgraphs.

Here is a powerpoint representation of the PlasmoDB workflow, a large workflow developed by EuPathDB to build the PlasmoDB.org database. Each slide in the presentation is a graph. The first slide is the root graph.

Here is a sample graph: getAndAnalyzeSNPs.xml

Here is a sample step class:

Additional Features

Graph Features

- conditional execution of steps
- changing the graph after the workflow has begun
- support for incremental builds
  - global steps
  - subgraph references
  - grafting

Runtime features

- test mode
- offline steps
- undo
- failure recovery
- load balancing

EuPathDB specific features

- resource steps
- tracking analysis methods

Information flow in the graph

Information is shared among steps in several ways:

- constants within the local scope of a graph
- global constants
- parameter values passed down through subgraph calls
- files in a shared directory structure

- your database

In addition, step class perl code gets a handle on these two property files:

- `steps.prop`
- `stepsShared.prop`

Running modes

ReFlow supports three running modes:
- test mode
- run mode
- undo mode

Downloading ReFlow

ReFlow, install, FgpUtils, DJob, CBIL

Installing ReFlow

Use the GUS Installer to install ReFlow. Instructions for downloading and using the installer are here: http://www.gusdb.org/wiki/index.php/GusInstaller. Once you have installed it and configured your environment, run the following command to install ReFlow:

- `> bld ReFlow`

If this is the first time you are ever using GUS, this step will fail. Copy the default gus.config file as specified in the error message. Run the above command again to build ReFlow.

You will also need a config file to control ReFlow's logging. Place the following text in a file called `$GUS_HOME/config/log4j.properties`

```
log4j.rootLogger=ERROR, CONSOLE
log4j.appender.CONSOLE.target=System.err
log4j.appender.CONSOLE=org.apache.log4j.ConsoleAppender
log4j.appender.CONSOLE.layout=org.apache.log4j.PatternLayout
log4j.appender.CONSOLE.layout.ConversionPattern=%-4r %d [%-5p] %C:%L - %m%n

# only display information under WDK domain
log4j.logger.org.gusdb=DEBUG
log4j.logger.org.apidb=DEBUG
```

Installing ReFlow on your compute cluster

ReFlow has support for these queueing systems:
- SunGridEngine

- PBS
- LSF

ReFlow additionally uses its own job distribution system called DistribJob. It is bundled in the ReFlow software download (DJob). It is a layer on top of the queuing system.

System administration tasks

1. install this software:
    a. java
    b. ant
    c. perl
    d. bioperl?
2. install bioinformatics software
    a. tba
    b. take notes on special cases only
    c. prepare a list of all the paths to the software installed
3. set up a directory in which all the data can go (eg /genomics/binf/scratch)
4. provide the following information:
    a. clusterServer (name of the master server)
    b. clusterBaseDir (a directory in the master's files system in which all data can go)
    c. clusterQueue (name of the queue to use, if any)
    d. nodePath (a working dir on each node that has lots of space and that will get cleaned up after job is done)
5. provide accounts on the cluster server

Other tasks

- install ReFlow on the compute cluster master server
    - follow the same approach as installing it on the local server
    - note: the programs that ReFlow runs on the compute cluster do not access any database. but, (for now) ReFlow still requires an Oracle driver in $GUS_HOME/lib/java/db_drivers to compile.
- install these additional projects:
    - DJob
    - CBIL
- if the cluster master is remote to the server ReFlow is running on, ReFlow will need to access it through SSH. Be sure an ssh agent is running so it can get there without needing to authenticate.
- set these properties in stepsShared.prop:
    - clusterBaseDir
    - clusterServer
    - clusterQueue
    - nodePath
    - nodeClass
        - DJob::DistribJob::SgeNode
        - DJob::DistribJob::PbsNode
        - DJob::DistribJob::LSFNode
- set properties in steps.prop indicating the paths to the bioinformatics software
    - what do they use as a template?

Running the test workflow

Local

need to cover here:
- copy installation from TestFlow project
    - samples/config/
    - samples/gus.config
    -
- including setting up the db
- running the flow to make sure things are working
- troubleshooting?

Compute cluster

need to cover here:
- copy installation from TestFlow project
    - samples/data/seq.fasta
- set values in steps.prop and stepsShared.prop

Creating a workflow

Creating a ReFlow workflow is an iterative process. The pieces of the puzzle that must come together are:
1. writing step classes in Perl. These are reusable modules that do work. Any type of task you want done is wrapped in a step class. For example, you might have a step class that runs BLAST.
2. writing the graph in XML. The graph has steps that depend on each other and can also call subgraphs. Each step is a parameterized call to a step class. Constructing the graph typically includes:
    a. Writing low-level graphs that are re-usable modules.
    b. Writing higher-level graphs that may call the lower level ones
    c. Writing a root graph that is the top-level of the graph nesting.
3. Setting up a home directory for a run of the workflow. This includes writing a number of configuration files.
4. Compiling the graph
5. Running the graph in test mode
6. Finally, after you have iterated on the above process, and completed development of the graph, running it for real.

Writing Step Classes

It may be that you already have an existing library of step classes, in which case you will only need to write step classes for new tasks you want that are not in the library.

Step classes are written in Perl and are subclasses of SUPER-CLASS-LINK-HERE. Here is a sample "hello world" step class that prints a message passed to it. LINK HERE.

The purpose of a step class is to:
- Declare its expected input parameters. Any call to the step class in the graph XML must provide

parameters that match this signature.
- Get a handle on the input parameter values at runtime.
- Validate the input parameters

Creating a workflow home directory

The workflow runs out of a *home directory*. Make a directory which will be your workflow's home. One pattern that might work for you is:
```
workflows/
        my site/
                my_sites_version/
```

where my_site is the name of the flow and my_sites_version is its version.

In this document we refer to your workflow home directory as my_workflow_home.

Setting up config files

The config files live in my_workflow_home/config. To get started, make that directory.

This directory contains much valuable information. You should consider having it under version control (SVN, CVS, etc). If you do, be sure that the repository is secure, as these files may contain sensitive information such as login/password info and file paths.

Files needed before you can compile a graph

The workflow.prop file

This file provides the workflow controller its most basic configuration information. It uses standard property file syntax.

- name=

    - the name of your workflow. This can be an arbitrary name, but typically is the name of your project.   It is only used internally by the workflow. Workflow steps do not have access to this value, so it does not have to agree with any project name that might be using.

- version=

    - the version of your workflow. This can be an arbitrary name, but usually reflects the version of your project. For example if your database is version 5.3, this might be "5.3-test" or "5.3-prod." It is not recommended for you to use "5.3" because you may have multiple workflows for a given version of your project, such as "test" and "production." The name and version together *must be unique* across all workflows that will run on a single database instance

- workflowXmlFile=

    - the root graph xml file

- workflowTable=

- the Workflow table you installed in your database as part of ReFlow installation

- workflowStepTable=

    - the WorkflowStep table you installed in your database as part of ReFlow installation

- workflowStepTrackingTable=

    - the table you installed in your database as part of ReFlow installation that tracks which rows in the application database were written to by workflow steps. In the GUS system, this would be ApiDB.WorkflowStepAlgInvocation

The rootParams.prop file

This file contains values that are passed in as parameter values to the root graph XML file. It uses standard property file syntax.

The steps.prop file

Use properties in this file to provide values to steps that are contingent on the environment in which the workflow is running. Values in this file are available to step classes when a particular step or type of step runs. An example would be the location in the file system of a particular executable program run by a step.

The stepsGlobal.prop file

Use properties in this file to provide values needed by global steps and that are contingent on the environment in which the workflow is running.

The stepsShared.prop file

Use properties in this file to provide values that are generally needed by step classes and that are contingent on the environment in which the workflow is running. Values in this file are available to three types of files:

- graph xml files, as macros using the @@property_name@@ syntax
- resource xml files, as macros using the @@property_name@@ syntax
- step classes, using the getSharedConfig() method

The resources xml files may also use the following macros, whose values are supplied by the workflow when it runs:

- @@dataDir@@

- @@gusHome@@

The loadBalance.prop file

This file controls load balancing during the run of the workflow. In the workflow graph xml files, different steps may be given different `stepLoadTypes=` values. These are arbitrary (single word) tags that designate a step as exerting one or more type of load on the hardware and software resources used by the workflow. For example, `stepLoadTypes=` might be set to "`computeCluster`" or "`database`" to indicate that the step uses those resources. The value be a comma delimited list such as "`sequenceTable,featureTable`".

In `loadBalance.prop` put the names of those tags, and give each a numeric value to indicate how many of those steps can run at one time. This way you balance the load.   You are required to have at least one line in the file indicating the total number of steps that can run at one time, like this:

```
total=10
```

If you have more tags, give each its own line:

```
total=14
database=8
computeCluster=9
sequenceTable=4
featureTable=5
```

Files required before you can test a workflow

The gus.config file

The workflow gets its database connection from the gus.config file (`$GUS HOME/config/gus.config`). This connection is used for both the application data and for the workflow's internal database.

The initOfflineSteps file

A file containing a list of step names (one per line). These files will be set to OFFLINE the next time the workflow starts up. This is identical to calling the workflowstep command with the -f option. The reason to use this init file is to prevent the workflow from running them before you have a chance to run workflowstep. To put these steps back ONLINE, use the workflowstep command. You can use the -f option and point it to this file. Also remove the steps from this file. NOTE: this file must exist, even if it is empty. Do not remove the whole file.

The initStopAfterSteps file

 (steps to pause after (e.g. to inspect DB state, etc.))

Note: if you want to stop after a subgraph is complete be sure to put the stop after on the return. For example: global.NRDB_RSRC.return

The loadBalance.prop file

 (configure load balancing)

Files required before you can run a workflow for real

The alerts file

This file controls email alerts sent when steps are done. (They are not sent when a step fails or in test or undo mode). The format is two columns, tab delimited, and there can be one or more rows. The first column is a perl regex (do not include the /'s) that will find the full name of a step or steps to send an alert for: the second column is a comma delimited list of email adresses. Here is an example that sends an alert on all Pvivax steps that end in makeDataDir:

```
pvivax.+makeDataDir$           joe@blotto.com, sue@flamers.org
```

See the section Testing Email Alerts to learn how to test this file before running with it.

Graph XML files

Constructing a single graph file

Input parameters

Constants

Steps

IncludeIf/ExcludeIf

load balancing

Calling subgraphs

Subgraph references

Compiling a graph

Checks that:

- the graph xml is valid XML

- that dependencies are correct
- that constants are correct
- that macros are correct
- that graph names are correct
- that variables are correct

You can "compile" your graph to test that all the variables, constants and subgraph references are valid:

```
$ workflow -h /files/cbil/data/cbil/TrypDB/wftest -c
```

An alternative way to do this, which also provides a detailed list of all the steps in the graph is:

```
$ workflowXml -h /files/cbil/data/cbil/TrypDB/wftest
```

Types of errors

In the following example, a variable has a problem. Go to the step mentioned and see what is wrong with the variable.

```
$ workflow -h /files/cbil/data/cbil/TrypDB/wftest -c

Parameter 'genomeExtDbRlsSpecList' in step 'common.InsertGenegenomicsequenceWithSql' includes
an unresolvable variable reference: '$$genomeExtDbRlsSpecList$$'
```

In this example, the mentioned step is calling a subgraph that expects the `genomeExtDbRlsSpecList` parameter, but the caller is not providing a value for it

```
$ workflow -h /files/cbil/data/cbil/TrypDB/wftest -c
Graph "compilation" failed.  The following subgraph parameter values are missing:

  File giardiadb/giardiaWorkflow.xml:
      step: common
          > genomeExtDbRlsSpecList
```

Global graphs

Use a *global graph* to make steps visible for dependency to steps anywhere in the workflow.

A typical application would be a step that many steps throughout the graph depends on --and which may need to be undone regularly for iterative build-- but which cannot be local to their subgraph because they are in reused subgraphs. (If the global step were in that subgraph it would be executed multiple times, each time the graph is reused.)

Any step in the workflow may use `<dependsGlobal>` to declare a direct dependency on any step in a global graph, with these rules:

- all `<dependsGlobal>` elements in a `<step>` or `<subgraph>` must come after any `<depends>` elements.
- In `<dependsGlobal name="xxxx">` xxxx must refer to the name of a step in the global graph

In the root graph, a subgraph may be declared to be global by using `<globalSubgraph>` instead of `<subgraph>`, with these rules:

- `<globalSubgraph>` is only allowed in the root graph
- there may be only one `<globalSubgraph>` and it must be the first step in the root graph.
- Unlike the `<subgraph>` element, the `<globalSubgraph>` element does not support `<depends>` or `<dependsGlobal>`. That is, the step that calls the global subgraph may not have any dependencies. (Inside the global graph there may be dependencies, like any other graph.)
- in the root graph, no step can depend on the `<globalSubgraph>`, ie, its name may not be referred to in a `depends=`
- A global graph XML has the same rules as a regular graph, except that a global graph may include `<globalConstant>` elements, with the following rules:
  - While a `<constant>` is visible only within the local XML file, a `<globalConstant>` is visible throughout the workflow
  - Use global constants sparingly, as they can lead to spaghetti. Typically, only steps that have a `<dependsGlobal>` should refer to them.
  - In a graph XML file, all `<globalConstant>` elements must come before all `<constant>` elements.
  - `<constant>`, `<param>` and `<globalConstant>` elements may embed variables referencing `<globalConstant>` elements.
  - but `<globalConstants>` may not reference `<contstants>`
  - it is strongly recommended that globalConstants are given names that begin with `global_`, eg, `global_NRDBOutputFile`. This way they will be recognizable where they are used.

Subgraph references

A `<subgraph>` element has an `xmlFilename=` attribute. The value of this attribute is typically a hard-coded xml file name. If the value of this attribute is a variable, then this `<subgraph>` element is a "subgraph reference." This is analogous to a *method reference* in a programming language. The pattern of its use is to embed in a standard graph a call to a graph that varies depending upon the context in which the standard graph is called. It is particularly powerful if elements of the standard graph depend on the subgraph reference. In this case, using a subgraph reference is critical.

Templates

Use templates to avoid cutting and pasting repeated steps within a graph xml file. This is helpful if there is one or a series of steps that get repeated within a file, say, once for each organism or sample. When you cut and paste the repeated steps and adjust their values for the particular case, you may introduce mistakes. Also, if the series of steps is long and the number of repeats is many, the file can grow too large to handle.

To see templates in action, try this command:

```
% workflowResolveMaster \ $PROJECT_HOME/ApiCommonWorkflow/Main/lib/xml/workflowMasters/testMaster.xml \
$PROJECT_HOME/ApiCommonWorkflow/Main/lib
```

Details:
- the templating is all done in `$PROJECT_HOME`
  - the ReFlow engine is unaware of templating. All workflow .xml files are fully resolved by the time they get to `$GUS_HOME`
  - the resolution is performed by `build.xml` (like we do for the perl object layer), when the `bld` command is called.
- there are three directories in `$PROJECT_HOME` that pertain to templates:
  - `../lib/xml/workflowMasters`
    - this holds master .xml files

- ○ `../lib/xml/workflowTemplates`
  - this holds template .xml files
- ○ `../lib/rtc`
  - this holds .rtc files
    - rtc stands for "reflow template configuration"
- master, template and rtc files are always stored in these directories so that they can be referred to with relative names.
- a *master .xml file* looks just like a workflow .xml file, except that it has `<templateInstance>` elements (described below). For example `plasmodb.xml` would become a master, since it will need to instantiate templates.
- a *template .xml file* looks just like a workflow .xml file, except that:
  - ○ it does not have <parameter> elements. (it is not a real subgraph.)
  - ○ it has embedded template macros. Maybe they look like this: `[[macro_name]]`
- a master .xml file instantiates a template like this:
  - ○ `<templateInstance name="p_falciparum"  xmlFile="organismTemplate.xml" rtcFile="plasmodbGenomes.rtc"/>`
- for example, `workflowMasters/plasmodb.xml` would have one `<templateInstance>` for each organism
- the master .xml file also has a new `<templateDepends>` element. It is just like a `<depends>` element however it also specifies the name of a template file in which the step will be found
  - ○ for example `<templateDepends name="LoadGenome" xmlFile="organismTemplate.xml"/>` might be used to connect the comparative genomics step to all xxxLoadGenome steps inside of templates.
- a *workflow .rtc file* is a property file with sections corresponding to `name=` in the `<templateInstance>` like this:
  - ○ `>p_falciparum`
  - ○ `some_property=some_value`
  - ○ `another_prop=another_value`
  - ○ `etc`
  - ○ `//`
  - ○ `>p_yoelii`
  - ○ `etc`
- the workflow.rng file (which specifies the xml schema) is updated to include the two new template elements.
- the ReFlow command `workflowResolveTemplate` takes the info that is found in a `<templateInstance>`. It does this:
  - ○ validates the template xml against the .rng schema
  - ○ returns an XML fragment corresponding to the body of the `<workflowGraph>` element, with all the macros resolved.
- the ReFlow command `workflowResolveMasters` resolves a master file. It does this:
  - ○ validates the master against the .rng schema
  - ○ scans the file
    - for each `<templateInstance>` element in the master, calls `workflowResolveTemplate` to expand the template instance in line
    - it expands each `<templateDepends>` element it finds.
  - ○ writes the output to standard output
- as a first cut, build.xml will always perform the template resolution for all found master files.
  - ○ it cannot use the master's time stamp to detect staleness because a master can also be stale if any of its templates are stale
  - ○ as a second cut, in Ant, it could compare the time stamp of the generated workflow .xml file with its master and all template files in the templates directory

Nested data directory structure

Files produced and consumed by steps are stored in a common directory structure. A preferred design for this directory is to have its structure mirror the nesting of subgraphs. This is accomplished by:

- each graph defining an input parameter called `parentDataDir`
- for the root graph, the value of parentDataDir is provided by the `rootParams.prop` file. This is the root of the directory structure
- each graph calls a `makeDataDir` step that creates a `dataDir` which is a subdirectory of its `parentDataDir`, named for this graph.
- recursively, each graph passes its `dataDir` to its subgraph as their `parentDataDir`

Steps within the same graph can use their graphs `dataDir` variable to refer to each other's files. A best practice is to use a constant for these file names to avoid file name mistakes

The root of this tree is in workflow home/data. The WorkflowStepInvoker superclass makes this available to all steps, via the getDataDir() method, which is relative to the root.

Resource steps

A set of steps in the workflow called *resources* acquire data from outside the workflow and make it available to other steps in the flow either as files or in the database. Or they may simply load the data into the database for use after the workflow is complete. The design pattern for a resource step is:

- a resource is defined using a <resource> element in a resource XML file. For example, datasources/plasmodb/pvivax.xml
- a resource XML file may contain many <resource> elements.
- a <resource> has a unique name that is an index into the resource XML file.
- a <resource> has a version
- the resource name and version are used in the database to populate the GUS tables SRes.ExternalDatabase and SRes.ExternalDatabaseRelease
- in the workflow graph, each resource is processed by a call to a subgraph called LoadResources.
  - LoadResource steps that are in a subgraph should all be in one resources XML file
- each step in the LoadResources subgraph is parameterized by
  - the resources XML file containing the resource definition
  - the name of the resource
  - the subgraph's `dataDir`
  - the `globalDataDir`
- the <unpack> and <pluginArgs> elements of a <resource> can use these macros:
  - `@dataDir@`
    - allows <unpack> and <pluginArgs> elements to share files
  - `@globalDataDir@`
    - allows <unpack> and <pluginArgs> elements to see files produced by global steps
  - and properties from the `stepsGlobal.prop` file. (eg `@soVer@`)
- if <resource> elements within a resources XML need to share files they must navigate the nested directory structure to find the needed file, for example `@dataDir@/../IEDB/iedb.fasta`. This is a workaround and should be avoided.
- with the exception of version= always use macros for a version, eg, `%RESOURCE_VERSION%` or `@goVer@`. The latter can be defined in `stepsGlobal.prop`.
- use a the `parentResource=` attribute of the `<resource>` to indicate that this resource is adding more data into a parent resource's dataset. In this case, the child resource does not get written into the DataSource table, and its data uses the parent's external_database_release_id. The `<pluginArgs>` element gets

access to the parent's extDbRlsSpec info by using the `%PARENT_RESOURCE_NAME%` and `%PARENT_RESOURCE_VERSION%` macros.

- the get and unpack phases of a resource might produce files that are consumed by dependent analysis steps in the graph XML. In order for test mode to work properly, the resource step must produce dummy files for these files. force that by using a `<getAndUnpackOutput>` element

Displaying Data Sources on a website

When the workflow loads a resource it records minimal information describing it: only its name and version. The `<resource>` element contains extensive provenance and descriptive information. This is stored in the ApiDB.DataSource table (and its friends). The information is placed there by the Tuning Manager, which reads the resources XML file from SVN (trunk). It only includes information for resources that are found in SRes.ExternalDatabase.name. This allows the resources XML file to accrue new resources after the flow is complete without those being seen on the website.

Testing the syntax of a resources xml file

Use the `validateResourceXml` program to compare your xml against the RNG schema definition `ApiCommonData/Load/lib/rng/resources.rng`.

Here is its usage:

```
$ validateResourceXml

usage: validateResourceXml -f resources_xml_file
```

Tracking Analysis Methods

Methodology information describing tools used in the workflow (eg, BLAST, Mummer) is stored by the workflow in the ApiDB.AnalysisMethod table (and friends). It is placed there by an InsertAnalysisMethod step. This step typically depends on the completion of the step that loads the results of the analysis. Any tool argument values that are recorded should be placed in a constant so they can be shared between the step that actually calls the tool and the step that records the methodology.

Best Practices

- if you change the input parameters in a subgraph, be sure to update all steps that call that subgraph, in all workflow XML files in the code base

Compiling a workflow

Testing a workflow

Once your workflow compiles, run it in test mode. The purpose of test mode is to exercise the step classes. It also gets you familiar with running a workflow and using the tools available. Running a workflow in test mode is

very similar to running in real mode.

For each step, it tests the following:
- that the step class exists
- that the graph is passing to the step class all expected parameters
- that the properties expected by the step class exist in the `steps.prop` or `stepsShared.prop` files
- that input files expected by the step class exist, ie, that upstream steps properly created them. (The files created in test mode are empty, but they have the correct file name.)

Before you run in test mode:
- set up needed config files

    - in `workflow.prop`, be sure your projectVersion number has some kind of test suffix, like "1.0-test." This way you will be able to use "1.0" for your real workflow run.

    - in `stepsShared.prop`, do the same thing. This ensures that test data files the workflow deploys to download site directories and other locations keyed on projectName and projectVersion will not collide with your real data files.

- set up gusdb config file

    - This is needed to store the workflow state

    - Running in test mode does NOT write to the application database

    - Modify $GUS_HOME/config/gus.config to contain the following database configuration values (you have to request them from a system or db admin):

        - dbVendor: probably "Oracle"

        - dbiDsn: perl DB connection info

        - jdbcDsn: java DB connection info

        - databaseLogin: DB username

        - databasePassword: DB password

    - You do not have to fill in any other fields to complete workflow test

To run in test mode, use this command:

```
$ workflow –h /files/cbil/data/cbil/TrypDB/wftest –t
```

Notes on test mode:
- when running a workflow you may find it convenient to have four terminal windows or tabs open, each one named as follows (or similarly):
    - **workflow**: a window which runs the workflow command (in the foreground)
        - it is best to run this command in `screen` to avoid unexpected termination. (Termination won't harm the workflow but will slow you down)
    - **log**: a window which runs tail -f my_workflow_home/logs/controller.log so you can watch the progress of the workflow
    - **steps**: a window in which you have cd'd to my_workflow dir and in which you can easily run workflow -h my_workflow_home -s1 FAILED to locate failed steps. Once you have fixed any steps, use the workflowstep command to set them to ready
    - **files**: a window which is cd'd to the directories that contain your graph and step class files, and in which you fix failed steps

- messages about Exclude indicate that steps are excluded from the graph. Either that step has excludeIf=true or includeIf=false
- ignore this error
  - /EuPathDB/workflows/apitest/1.0-test
    could not find ParserDetails.ini in /usr/lib/perl5/site_perl/5.8.5/XML/SAX
- to handle initial failures:
  - they are likely to be systemic, ie, errors in stepsShared.prop
    - if you fix property files, you need to change the individual FAILED step to READY or reset the whole test flow (see Resetting a workflow below)
  - if you change the graph.xml file (eg, correct a step class name), you need to bld and then restart the flow engine

1.
2. workflowstep -h /files/cbil/data/cbil/EuPathDB/workflows/apitest/simple-test -p '%_RSRC' ready
3. What happens if i kill the engine while my test is running?
   a. steps that are running continue to run safely
   b. they successfully update the workflow engine database
   c. just restart your workflow!
4. instead of grepping the controller.log for FAILED steps, use this command (the log may contain old info):
   - workflow -h my_workflow_home -s FAILED
5. once you fix a step, you can change its status from FAILED to READY with this command. (Use the full step name path):
   - workflowstep -h my_workflow_home -p step_name ready
6. if you made a fix that will correct a set of steps you can set them all to ready by using a pattern for the step name. use % as a wildcard. The following example finds any step with Nrdb anywhere in its full step name path
   - workflowstep -h my_workflow_home -p %Nrdb% ready
   - note that it is ok if your pattern finds steps that are not FAILED. The output of the workflowstep command will show you some warnings like this, which you are free to ignore:
     - Warning: Can't change PbergheiPostLoadGenome.genomeAnalysis.blastxGenomicSeqsNrdb from 'DONE' to 'READY'
7. To find out if the workflow is still processing steps, or if it is blocked by FAILED steps, run this command to find ON_DECK steps. (If there are none, then it is stalled, and you must fix FAILED steps to give it steps to process):
   - workflow -h my_workflow_home -s ON_DECK
8. if fixing a FAILED step involves correcting the step's graph XML, then you will need to restart the controller so it can pick up the new XML files (don't forget to do a bld first). Just kill the controller and then restart it.
9. if fixing a FAILED step involves correcting a step that the FAILED step depends on, then you will need to UNDO the dependee.

Reseting a test workflow

You can blow away your test workflow by reseting it.

10. reset the workflow, if there is any old one of this name in there
    a. you can reset a test workflow (but not a real one)
    b. CAUTION: this will WIPE OUT your test workflow's home dir (not config/) and the workflow tables
    c. might need to manually clean up cluster dirs
    d. might need to manually clean up download dirs

The workflow tables

The workflow controller stores its persistent information in two tables:
- Workflow

- WorkflowStep

You can look at your workflow and its steps in these tables. Do not write to them unless you know *exactly* what you are doing, because you can easily hose your workflow.

Changing a graph after the workflow has begun

A graph may be changed after a workflow has begun with the rules that follow.

Rules you must follow *before you install your new graph into* $GUS_HOME:

1. stop the workflow
2. no step may be in the RUNNING state. wait until all steps are done running.
3. no step may be in the FAILED state. use the workflowstep command to set them to READY while the workflow is not running.
4. an existing step may be changed (or deleted) *only* if it is in the READY or ON_DECK state
5. new steps can be added to a graph in arbitrary ways with these rules:
   a. anywhere the graph is called in the workflow, the return step must be in the READY or ON_DECK state. If it is not, UNDO it.   There are three cases:
      i. the only steps that depend on the return step are themselves return steps. In this case, the graph is at a leaf. UNDOing the return is a lightweight operation
      ii. some steps that depend on the return are not returns, but all are in the READY or ON_DECK state. In this case the graph is not at a leaf, but UNDOing the return is lightweight because it will not involve UNDOing any real work, as none of its dependents are DONE.
      iii. like 2 but some of the non-return dependents are DONE. This is not lightweight, as it involves UNDOing real work.
   b. existing steps that are made to depend on new steps must comply with rule 3

Rules for changing the XML:
6. the input parameters of a graph can be changed or deleted, and new ones added with these rules:
   a. calls to the subgraph *may* be in the DONE state
   b. all callers of the subgraph must be changed to correctly match the new expected parameters
   c. no changed or deleted parameters are consumed by a step that does not comply with rule 3

11. to delete a step from the graph (or exclude it using includeIf or excludeIf):
    a. use the workflowstep -l option to find the state of the step
    b. if the step is DONE, undo it
    c. stop the controller
    d. if the step is FAILED, use workflowstep to set the step to ready
    e. delete the step from the graph (don't forget to do a build)
    f. start the controller
12. rules for changing the XML graph
    a. no steps may be in FAILED or RUNNING state
       i. for RUNNING steps, wait till they complete
       ii. for FAILED steps
          1. stop the controller
          2. do any cleanup of the step that is needed
          3. use workflowstep to set the step to ready
    b. you may not change steps that are DONE
       i. use UNDO to set DONE steps to READY
    c. if may not change the graph after entering UNDO mode

       i. you may change the graph and then enter UNDO mode, but after the first time you run with -u, you may make no changes until the UNDO is complete

Testing email alerts

If you provide the file my_workflow_home/`config/alerts` (see the Configuration section above), the workflow will send email alerts when the specified steps are done.

Errors in this file will prevent the alerts from being sent. If you want to add alerts to the file, test it first:

1. make a temporary alerts file with the alert or alerts you want to test
2. use the `workflowAlertsTest` command to test that file. It will show you which steps will generate an alert
3. append the tested alerts to the real alerts file in the config/ dir

Running the Test Workflow

Basic

Running the test workflow will demonstrate a working, usable workflow. The steps do not perform any meaningful actions, but they will show how workflow steps run in parallel, and how subgraphs can be called. To run the test workflow that is packaged with the ReFlow project, do the following (some of these steps may already have been completed):

1. Check out install, GusAppFramework, FgpUtil, and ReFlow from Subversion
2. Set up your environment (setting PROJECT_HOME and GUS_HOME)
3. cd $GUS_HOME; mkdir -p bin config lib/java/db_driver; cd -
4. Set up gus.config as in the "Testing a Workflow" section above and place in $GUS_HOME/config
5. Place the Oracle driver jar in $GUS_HOME/lib/java/db_driver
   a. On gattaca this can be found at
          i. /usr/local/home/oracle/GUS/gus_home/lib/java/db_driver/ojdbc5.jar
6. Create a log4j.properties and place it in $GUS_HOME/config
7. bld FgpUtil
8. bld ReFlow
9. mkdir testWorkflow
10.
11. cp -r ReFlow/TestFlow/data/config testWorkflow
12. workflow -h /<full_path_to_current_dir>/testWorkflow -t
   b. Before this step you may want to do the following:
          i. Change the workflow name (i.e. change
          ii. Reset the workflow (workflow -h testWorkflow -reset)
13. Confirm that the workflow completed successfully by reading the log
   c. Log file is at testWorkflow/logs/controller.log
   d. Step status/error messages are in testWorkflow/steps

Compute cluster

- seqs.fasta

Running a workflow

Running a workflow in real mode is very similar to running in test mode. The main difference is that the workflow does real work, rather then pretend to do it. The operative differences are...

Load balancing

Taking steps offline

Handling Step failure

Undo

Visualizing a Workflow

Visualization while Authoring

While authoring a workflow simply involves editing the workflow XML files in a workflow directory, it may be handy to see the workflow steps and subgraphs in a graphical, linkable format along the way. The Authoring GUI provides this functionality. To run the authoring GUI and see your workflow in graph format:

- Build ReFlow
  - `> bld ReFlow`

- Run `workflowHtmlGenerator` on your workflow XML directory
  - For usage, run the executable with no arguments
  - Sample use:
    - `> cd $GUS_HOME/lib/xml/workflow`
    - `> workflowHtmlGenerator -r . .html`
  - (if you put the html directory in the directory that contains the graphs, be sure to make it hidden)

- Run a simple webserver to serve the html files. Here are two examples:

- **A:** python:
  - open a new terminal (eg, in a tab)
  - `> cd $GUS_HOME/lib/xml/workflow/.html`
  - `> python -m SimpleHTTPServer 8080`
  - use ctrl-c to kill the server

- **B:** workflowGraphServer:
  - `> workflowGraphServer 8080 html`
  - (this program by default uses Python's SimpleHTTPServer module as above. It also offers an option to use Jetty, if installing Java/Jetty on your server is preferable)
  - use ctrl-c to kill the server

- On your local machine (eg, laptop), open a browser to <u>http://localhost:8080</u> (or whatever port you specified)

To edit a graph:
1. edit the associated XML file.
2. Re-run `workflowHtmlGenerater`.
   a. (It will only re-generate graphs for xml files that are changed.)
3. Reload the browser window.

Port Forwarding

The examples above use port 8080. However, you should probably use a different port, as 8080 might be in use. Choose a port that you think will not be used by others.

If you want to use a local browser (eg, on your laptop), you need to use port forwarding. To do so, choose a port that is not in use on your local machine and forward it to the port you chose on the server.

Windows

Here is an example in SecureCRT (Windows machines):
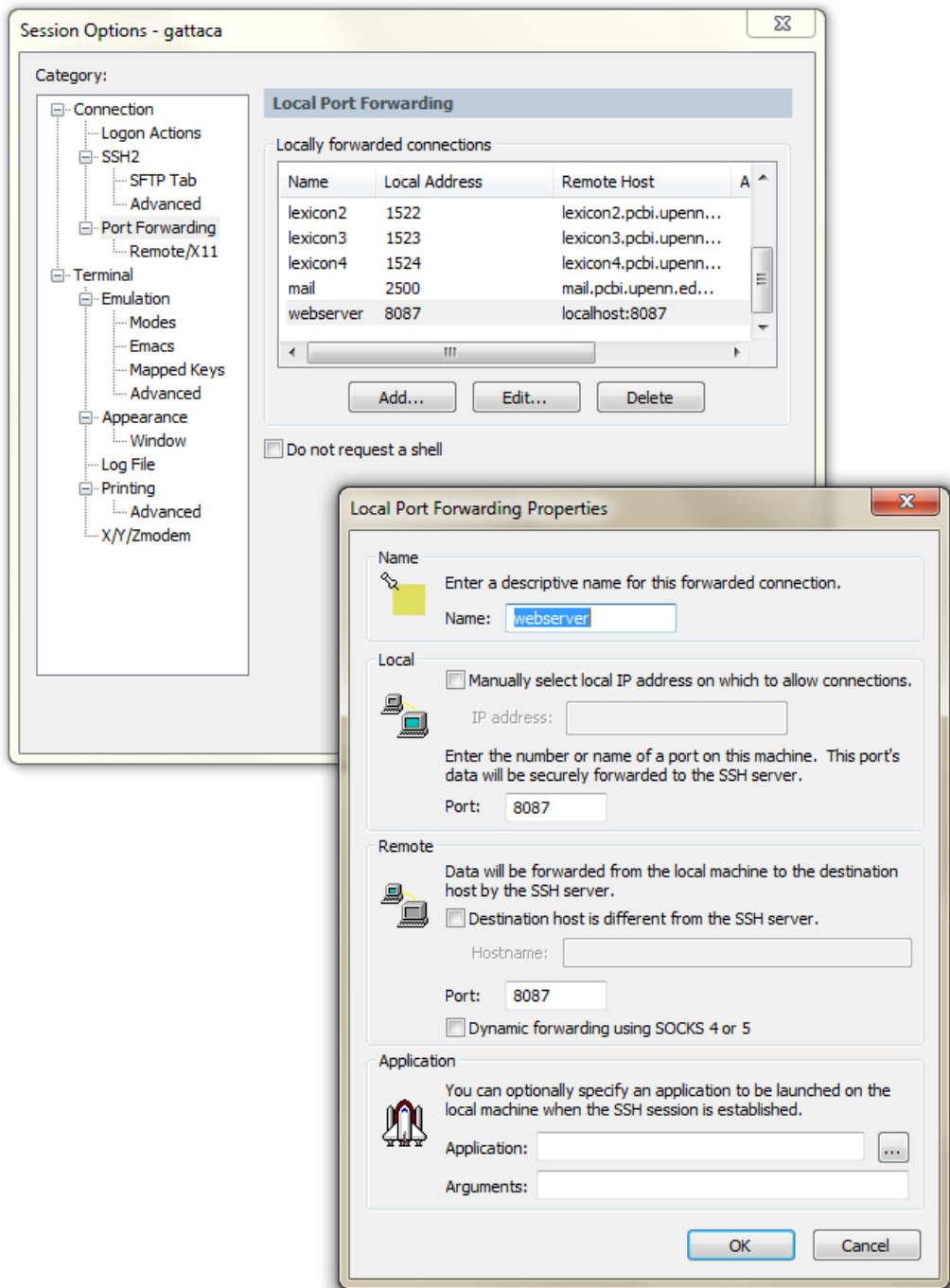
Unix variants

If you are using OS X or Linux, you can set up the port forwarding as a part of the ssh command you use to log in to the server. For example, if I (Brian Pitts) wanted to forward port 8080 on cassini to my laptop, I could run

ssh -L 8080:localhost:8080 pbr@cassini.pcbi.upenn.edu

GUS dependencies or assumptions

- in GUS/Workflow
    - WorkflowStepInvoker
        - move to WorkflowStep.pm
            - runPlugin
            - getUndoPlugin
            - getAlvInvIds
- in ApiCommonWorkflow
    - in Workflow.pm
        - move to WorkflowStepInvoker
            - getWorkflowDataDir
            - testInputFile
            - getInputFiles
            - getDataSource
            - DataSource.pm
    - resource acquisition
        - resource XML
        - manual delivery
        - plugins
        - external_database

Questions

- question: how do i pass files from one subgraph to another?
    - answer: the most recent ancestor of the subgraphs defines a <constant> that will be the name of the file. it is passed as a param to the graph that produces the file and as a param to the graph that consumes it
- question: how to define optional subgraph references? (e.g., global, isolatesResources)
    - answer: use includeIf/excludeIf. Note: for an excluded subgraph, the xml file name does not need to be a valid file name
- question: best solution to pass extDblsSpec from resource to analysis steps? (e.g., getAndAnalyzeChIps.xml, chipExtDbRlsSpec)
    - answer:
- what does this error mean:
  /home/weili1/gusApps/workflow/gus_home/lib/xml/workflow/plasmodb/plasmoWorkflow.xml:62:57:
  error: element "globalSubgraph" not allowed in this context
  Validation failed:
  file:/home/weili1/gusApps/workflow/gus_home/lib/xml/workflow/plasmodb/plasmoWorkflow.xml
- if i get this error, how do i find the problem:
    - Step
      'PreichenowiSpecificWorkflow.mapIsolatesToGenome.blastnIsolateSeqGenomicSeqs.loadSimilarities'
      has changed in the XML file while in the state 'DONE'
      old name:
      PreichenowiSpecificWorkflow.mapIsolatesToGenome.blastnIsolateSeqGenomicSeqs.loadSimilarities
      old params digest: 35360f49d5bfa4a6557c724a2a96d4f4
      old depends string: [mirrorFromCluster]
      old class name: ApiCommonWorkflow::Main::WorkflowSteps::InsertBlastSimilarities

      new name:
      PreichenowiSpecificWorkflow.mapIsolatesToGenome.blastnIsolateSeqGenomicSeqs.loadSimilarities
      new params digest: 8ac0d35a2280f17067e7fbaa6ccaa33c
      new depends string: [mirrorFromCluster]
      new class name: ApiCommonWorkflow::Main::WorkflowSteps::InsertBlastSimilarities
    - look in step.err at the arguments to workflowstepwrap to see what params were passed to the step, and see how they differ from their value now.

Notes

- put any notes here that you like. we can incorporate them into the document later, if that is helpful....
- in resources XML we have a new attribute of the <resource> element called parentResource=
- *never* hard-code a version. always use a macro.
- changing sharedSteps.prop when controller is running is dangerous because the changes won't be seen until the controller stops and restarts. by then step that consume the values might have already run. this is true even if the changes desired are intended for the resources xml file.

- Running a workflow: Consider running a workflow in 'screen to capture any error output on the command line.'

- If a step that runs a plugin FAILs, before you set it to READY, clean up what it wrote to the database by calling the plugin undoer:

  - `ga GUS::Community::Plugin::Undo --plugin XXXXXX --algInvocationId ## --workflowContext --commit`

-

llsl

DistribJob Documentation

DistribJob – *A Distributed Job Controller*

Overview

DistribJob is a distributed job controller: it distributes elements of an input set to nodes in a compute cluster for processing, and merges the result.

More specifically, the *controller* runs on a *server*. It breaks a large *input set* into smaller sets called *subtasks*, and assigns the subtasks to compute *slots* on *nodes* (machines) as they become available. On the node, it runs a *command* on the subtask's input, and merges the result into the *main result* on the server. It robustly tracks failures and can be restarted.

DistribJob does not do scheduling or load balancing. For a particular job, it is given a static set of nodes, each with a static number of slots. When a slot is vacated, DistribJob fills it with the next subtask.

DistribJob is written in Perl, using Perl objects and modules.

Caveat: DistribJob is academic software. It has been extensively used and tested, and is robust. However, its installation, documentation and usage are not brought to the level of industry polish. For example, some of the steps below forego thorough handholding, expecting instead for you to open a Perl file here or a test file there. None of the steps are hard. Just rough-cut.

Download

The DistribJob software is available for download from the GUS SVN repository. You will also need need the CBIL libraries also available from the GUS SVN repository.

Installation

Use the GUS Installer to install the DistribJob and CBIL software.

DistribJob comes in two modules: the DistribJob module itself, which is the core, and, the optional DistribJobTasks module, which includes a set of predefined bioinformatics tasks ("tasks" are defined

below). The latter are useful as samples, even if the particular tasks included are not relevant to your application.

To install either one, acquire the distribjob package (either through downloading or cvs). Then follow the instructions in the `DistribJob/readme.txt` file or `DistribJobTasks/readme.txt`. Installing DistribJobTasks will also install DistribJob (but not vice versa).

Testing your installation

DistribJob and DistribJobTasks include a set of tests. You are advised to run these before proceeding.

Basic local test

Once you have completed the installation, the first test is:
```
% distribjob $HOME/test/DistribJob/input/controller_local.prop 1 2
```

This test should work right out of the box if you installed your test/ directory into $HOME. Otherwise, edit `test/DistribJob/input/controller_local.prop`. Change the values of properties that refer to files in the test/ directory to reflect the actual location of test/.

The test should produce output that looks like this:

```
[sfischer@pythia ~]$ distribjob test/DistribJob/input/controller_local.prop 1 2
Reading properties from test/DistribJob/input/controller_local.prop
Reading properties from /home/sfischer/test/DistribJob/input/task.prop
Finding input set size
Input set size is 30
Subtask set size is 3 (10 subtasks)
Initializing server...

Initializing node 1...
Initializing node 2...
.
dispatching subTask 1 to node 1.1

dispatching subTask 2 to node 2.1
.
subTask 2 failed

dispatching subTask 3 to node 2.1
......
subTask 1 succeeded

dispatching subTask 4 to node 1.1

subTask 3 succeeded

…

subTask 9 succeeded
```

```
Cleaning up nodes...

Failure: 2 subtasks failed
Please look in /home/sfischer/test/master/failures/*/result
After analyzing and correcting failures:
  1. mv /home/sfischer/test/master/failures /home/sfischer/test/master/failures.save
  2. set restart=yes in test/DistribJob/input/controller_local.prop
  3. restart the job
```

The test purposely includes two subtasks that fail. This will give you the chance to practice correcting failures and restarting. The source of the problem is that two of the lines in test/DistribJob/input/inputset (which is the input for this job) have "oops" on them. To correct the problem:
- Edit that file, removing the oops
- Remove the master/failures directory
- Set the property restart=yes in controller_local.prop
- Start the job again.

Basic liniac test

This test is exclusively for UPenn's Liniac. To run it:
% cd $HOME/test
% liniacsubmit 2 2 $HOME/test/DistribJob/input/controller_liniac.prop
The immediate output is the message confirming entry to the queue.
Once the queue runs the job, its output (identical to that from the local test) will appear in $HOME/test/xxxxx where xxxxx is the log file starting with the queue's job id.

DistribJobTasks tests

The bioinformatics tasks included in the DistribJobTasks module also have tests. Run these in a similar fashion to the basic tests. You may need to edit the task.prop files to specify where your bioinformatics resources are.

Specifying your task

You provide a *task* to DistribJob (see configuration below). The task determines:
- The allowable type of input
- How the input is subdivided into subtasks
- How the server is to be initialized before the task starts
- How each node is to be initialized before the task starts
- What command to run on the node for each subtask
- How to merge the subtask results into the main result

Chose from the built-in tasks provided by the DistribJobTasks module, or code your own task.

Coding your own task (advanced)

To code your own task, you will need to write a subclass of DistribJob::Task. For samples, see:
- `DistribJob/lib/perl/DistribJob/SampleTask.pm`
- `DistribJobTasks/lib/perl/DistribJobTasks/*Task.pm`

You will also need to define the command that will run your subtasks on the nodes. The command must:
- Take as input a subset of the original input. It may restrict itself to only one element of input, but would probably be more efficient if it can handle a set. (This depends on the speed with which a single element is processed. The subtask should run for at least a few seconds to mitigate the overhead of passing data back and forth between the server and the node.)
- Accept full path names on all input files
- Write all temp files to the directory in which it runs.
- Terminate with non-zero status on any error condition that might need to be corrected.

Constraints on input

Following are constraints that apply to your task's input:
- The input must be conceptually an array. The controller will provide the task with the starting and ending indices of a subtask, and the task must provide a file or files that represent the associated elements for that range of the original input.
- The order of processing the elements must not matter
- You may not delete or add elements from or to the input when restarting a job.

Using built-in nodes

DistribJob distributes the subtasks to *nodes* (ie, machines in a cluster). You will specify how many compute *slots* each node has when you configure DistribJob (discussed below). The DistribJob package includes a growing number of built in types of node including:
- **DistribJob::BprocNode** is a node in a BPROC cluster.
- **DistribJob::SgeNode** for clusters utilizing the Sun Grid Engine.
- **DistribJob::PbsNode** for clusters utilizing PBS.
- **DistribJob::LocalNode** is a virtual node running on your local machine.

The LocalNode can be run on any multi-processor or even single-processor machine, where it is efficient to have more than one subtask running at a time.

Coding your own node (guru)

If your cluster uses a process control system other than one of these, you can still use DistribJob, but you need to write some simple code. `DistribJob::Node` is the object which represents a node. Its main

purpose is to specify how to communicate between the server and node. The details of particular types of nodes are specified by subclasses of `DistribJob::Node`, which is what you will need to write. To learn how, use `DistribJob::BprocNode` and `DistribJob::LocalNode` as samples.

You may also want to help your user by providing a cluster-specific startup script. This script will submit a job to your cluster's queue, and then call `distribjob` (see Running below) when the job is ready to run. As a sample see `DistribJob/bin/liniacsubmit` which submits a job to UPenn's Liniac cluster. Also see `DistribJob/bin/liniacjob`, which is the script that runs. It in turn calls `distribjob`.

Setting up to run

The first step is to decide where you want your input directory, and where you want your master directory. Typically, these will go in a directory dedicated to this run of your task. Just make sure there is enough storage space to accommodate your results.
As an example, lets say you chose to run in $HOME/myrun, and that your task is creating a BLAST matrix (one of the provided bioinformatics tasks). Create your input directory, and copy the input file to it (in this case, a set of DNA sequences).
```
% mkdir -r $HOME/myrun/input
% cp myseqs.fsa $HOME/myrun/input
```

Configuring

Create two configuration files for your task (the standard place for them is your input directory):
- **`controller.prop`** contains the properties required by the controller. To see sample values required for this file, look in `DistribJob/test/input/controller_*.prop`. The properties are explained in detail in the full help for the distribjob command (`distribjob -help`). (You can name this file whatever you like, but `controller.prop` is the convention).
- **`task.prop`** contains the properties required by your task. If you are using one of the provided bioinformatics tasks, find sample task.prop files in `DistribJobTasks/test/*_input/task.prop`. In addition, the properties are defined in Perl files which subclass Task.pm. Look in `DistribJobTasks/lib/perl`.

Other resources

You may need to make other resources available to your task. For example, the provided BlastMatrixTask and BlastSimilarityTask make use of a database of sequences to BLAST against. This file may be very large, and so, you will not want to copy it into your input directory. In this case, you just specify the resource's location using the appropriate property in your task.prop file.

Running

The `distribjob` command starts the controller. Running it with no arguments prints its usage. Running it with `-help` prints its full help display.

You use different commands to run locally or to run on different clusters.

Regardless of how you run, the result of all the subtasks will be merged into `master/mainresult`.

Running locally

If you use `DistribJob::LocalNode` as your node type, you are running locally (ie, on your local server). You do not need to submit your job to a cluster queue; you just run `distribjob` directly. To do so, use this command (if, for example, you want to distribute across 3 virtual nodes):
```
% distribjob your_controller.prop 1 2 3
```

Running on UPenn's Liniac cluster

You are assumed to know how to use UPenn's Liniac cluster. To run `distribjob` on the Liniac, set the `nodeClass` property in your `controller.prop` file to `DistribJob::SgeNode`. Rather than calling `distribjob` directly, submit your distributed job to the Liniac's queue by calling `liniacsubmit`. Here is its usage:
```
% liniacsubmit nodecount minutes controllerPropFileFullPath
```

`liniacsubmit` produces as immediate output the standard queue submission report. If you want to run on 20 nodes and estimate your job will take 10 hours, use this:
```
% cd where_I_want_my_log/
% liniacsubmit 20 600 /my_inputdir/controller.prop
```

When the queue runs your job, it will place the job's log in the directory where you ran `liniacsubmit`. Check this log to see your job progress.

To see the status of your job on the queue, run:
```
% showq
```

Running on a different type of cluster

If you plan on running on a cluster type other than UPenn's Liniac, you or your administrator will need to provide commands that parallel `liniacsubmit` and `liniacjob`.

Handling problems

These are the kinds of problems you may encounter:

- **False starts**: annoying little errors that prevent your job from really running
- **Failures**: the subtask running on a node dies because of
  - reproducible problems
    - missing executables
    - missing files or directories
    - data errors in the input
  - non-reproducible problems
    - the node goes weird
    - the command goes weird
- **Hung jobs**: the subtask running on a node gets hung
- **Wrong number of nodes**: you realize that you want to add more (or take away some) nodes. To do this, kill and restart with the desired number of nodes (see below).
- **Need to kill**: you realize that you need to kill your job

False starts

These are errors in which the job immediately fails, and no work is dispatched to the nodes. The most common reason is that you have an error in a configuration file. The log should explain the problem although it might be a little less-than-obvious.

Once you have corrected the problem, you need to delete your master directory, and start again.

Failures

When a subtask running on a node fails, the log will report the failures and the job's files are copied to a directory in `master/failures/subtask_nnn/result`. Look carefully in all the files to determine the cause of the failure.

If the problem appears to be one that will happen reproducibly, then you need to correct the source of the problem. For example, you may need to correct your input file (but don't delete or add elements… this will mess up the indexing used by the controller). Or, you may need to provide missing data or executable files.

If the problem seems like a random flux of the cosmos, then you can defer correcting the source of the problem.

After you have handled all the failures, and when your job is no longer running, delete the `master/failures/` directory and restart the job (see below).

Hung subtasks

The directory `master/running` contains subdirectories for each running subtask. Hung subtasks will have subdirectories there whose subtask number should have long since come and gone.

If you detect a hung subtask, you will need to kill your job and restart (see below). You can either wait till the rest of the subtasks are complete or, if you feel that the hung subtask(s) is(are) using resources that you would rather have working for you, you can kill forthwith.

Need to kill

You may need to kill your job, either because you have hung subtasks, or because it turns out to be a virus bent on conquering the world. There are two steps you need to take:
1. kill the distribjob controller:
   a. kill it as soon as possible (without corrupting its results):
      i.   `% distribjob controllerPropFileFullPath -kill`
   a. kill it without interrupting running subtasks:
   i.   `% distribjob controllerPropFileFullPath -killslow`
1. If you are running in a queue, kill the job in the queue. (On UPenn's Liniac, use the `canceljob` command).

Restarting

To restart a job, change the `restart` property in the `controller.prop` file to `yes`, and start the job the same way you did the first time. Subtasks that are already complete will be skipped. (This is controlled by the file master/ completedSubtasks.log, which is a list of completed subtasks. If you need to redo a subtask that has already completed –at your own risk-- then you can delete its number from the list. But, remember its results might already be merged into the main result).

Strategies WDK: Overview
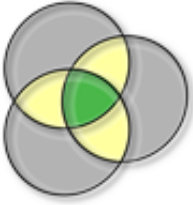
Strategies WDK Documentation

WDK Version: 2.0 (4/29/2011)

Please see the Strategies WDK Project Home for details about the project and useful links, etc

Chapters
1. **Overview**
2. Requirements
3. Installation
4. Creating a site
    a. Designing a model
    b. Configuring and customizing the view

1. Overview

Introduction
Background
The WDK Design
    The end user's perspective: Search Strategies
        Searches (a.k.a. Questions)
        Summaries and Strategies
        Records
    MVC: Model-View-Controller Architecture
        The Model
        The View
        The Controller

Introduction

The Strategies WDK is a framework for creating "data mining" genomics websites. It is a layer on top of your relational database. Supported DBMS platforms are Oracle and PostgreSQL. It is schema independent, which means it does not require the data in your database to be in a particular form (see Requirements for some restrictions).

Use the WDK to:
- define a coarse-grained data model (based on the tables in your existing database) that specifies the searches the user can run and the kinds of results he or she can get
- define a customized view of that data model using Java Server Pages

The data model that you define in the WDK is a *Data Transfer Object (DTO) layer*. A [DTO](#) is an object that brings together data that may come from many tables in the database. It is good practice to provide to a web site or other high-level data consumers objects at a coarse granularity. In the WDK the DTOs are called *records*. For example, a Gene record may bring together data from many tables that contain information relating to a Gene. Records in the WDK are *configured* in XML.

In sum, the WDK model lets you configure *records* and *searches* that return sets of them.

Background

The WDK evolved out of long standing development efforts at the [Computational Biology and Informatics Lab](#) (CBIL) at the [Penn Center for Bioinformatics](#) (PCBI), University of Pennsylvania.

> The WDK is written with these web technologies:
* RELAX NG compliant XML to define the functionality of the site (the model).
* Java Server Pages, JavaScript, AJAX and JQuery (and custom tags) to define the look of the site (the view).
* Struts to control the site (the controller).

The WDK is open source available under the Gnu Lesser GPL.

The WDK Design

The WDK helps you:
* make pages for the entities you want users to see (such as Genes)
* offer many different types of searches for those entities
* display and manage search results
* upgrade your site as your schema and data evolve
* offer data mining tools such as
    * set operations on results
    * sortable results
    * query history
    * data set upload
    * customizable data download

The end user's perspective: Search Strategies

The WDK uses a *Search Strategies* paradigm to organize a web site. End users are provided with a set of *searches* to chose from. They run a particular search by specifying values for the search's *parameters*. The result is a *summary* of the records found by the search. Each row in the summary offers a link to the full *record page.* This search automatically becomes the first step of a *Strategy*. The user can add steps to the strategy to refine the result. This [four minute tutorial](#) provides a good introduction.

Searches (a.k.a. Questions)

In the WDK we call the inquiries that users pose *searches* as opposed to *queries*. This is because in the

WDK searches return a set of records while queries in database systems to return rows. As we describe below, the WDK does use queries, but they are hidden from users.

Note: Internally, the WDK calls searches "questions." If you see "question" in the documentation or code, think "search." They are the same thing. The user thinks of it as a search, the developer as a question. (This is a legacy term that will likely remain.)

The 🔖home page of Plasmodb.org offers the user about 100 searches. Some can be run directly from the home page. But most are run from a dedicated dialogue page. The WDK lets you define searches and put them into different nested categories. This way you can offer your user expandable menus of searches which lead to individual search dialogue pages. The PlasmoDB search menu looks like this:
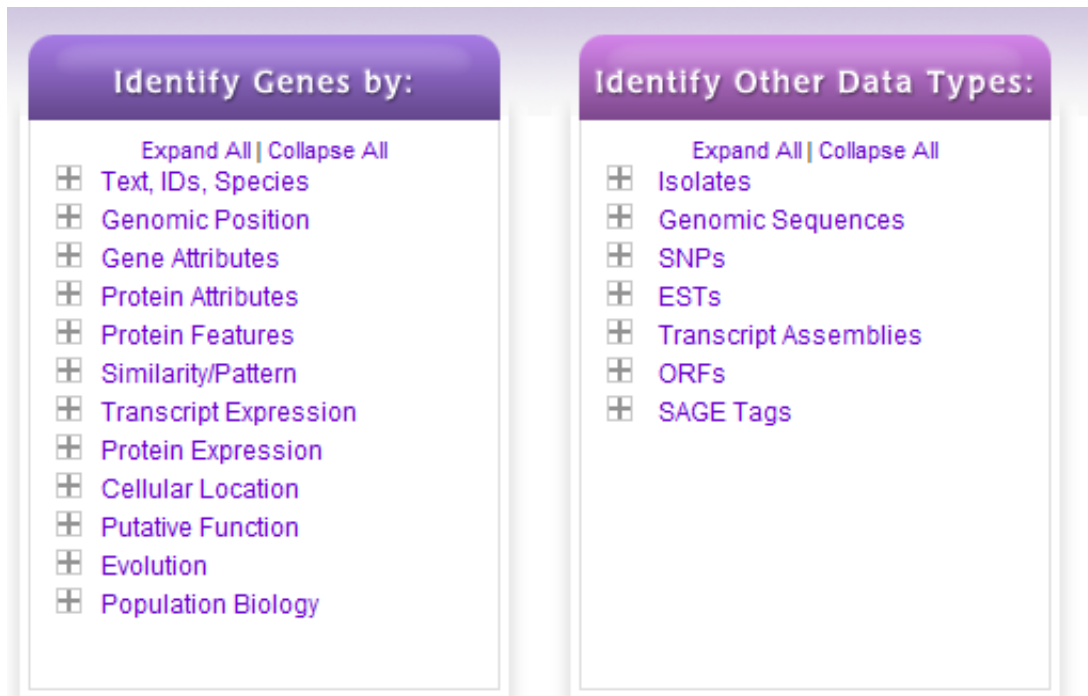
When the user chooses a search, he/she is brought to a search page, internally called a Question Page. On PlasmoDB.org, if the user selects the "Transmembrane domain" search from the nested menus, he or she will arive at the 🔖Transmembrane domain question page. On this page the user will fill in parameter values and submit the search.

It looks like this:

Summaries and Strategies

Once the search is run, the user is brought to a page showing a summary of the result. The summary displays a row for each record that matched the search. The first column is a link to the detailed record page. This is what it looks like:

At the top of the page is the Strategy Panel, and below it is the Summary Table showing the list of records that match the search. If the user adds a new search to the strategy, the Strategy Panel will look like this:
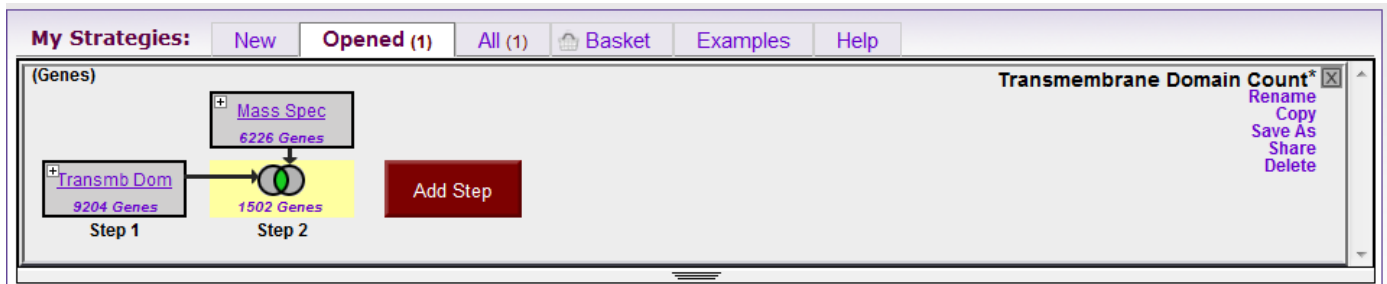
Records

When the user clicks on a record ID in the Summary Table they go to that record's page. In our PlasmoDB example, the user will go to the gene record. The record presents detailed information about the Gene, including:

- its organism, chromosome, name, fuction, source of annotation
- links to additional pages showing more details
- tables of information (e.g., Notes and Gene Ontology Assignments)
- graphics illustrating details of the record

Here is portion of PlasmoDB's Gene record page:

MVC: Model-View-Controller Architecture

The WDK uses the Model-View-Controller architecural pattern. This cleanly separates *what* we are describing (the Model) from *how* we display it (the View).

Here is an overview diagram:
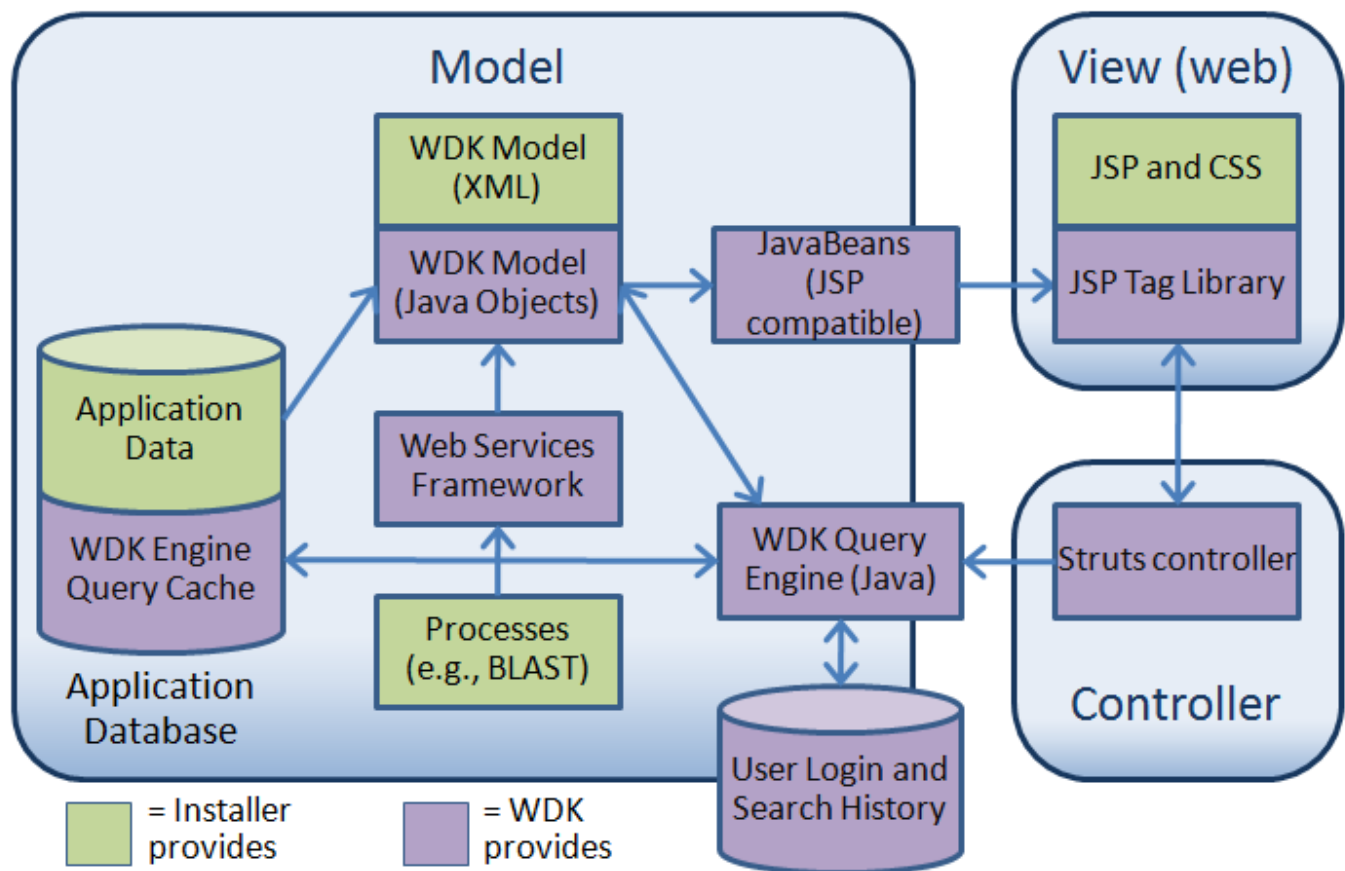
Scientific data is stored in an application database of any design (supported vendors are Oracle and PostgreSQL). The WDK's query cache can reside in that database or separately. The WDK Model is specified in XML and instantiated into Java objects that represent records and searches. The query engine uses the Java model (available to the front-end through a JSP-compatible JavaBeans API) to perform searches against the application database, or using a web services framework. The web view is primarily composed of JSP, JSP tag,sand CSS, and also uses JavaScript, JQuery and AJAX .

The Model

The WDK model describes the searches, summaries and records that will appear on the website. The model

is defined in XML files that you create (you may borrow from template model files supplied with the release). Creating a model is described in the section Creating a model.

In overview, the things you define in the XML model are:

- **Records**: the course grained objects that users will understand as the basic entities of the website. These are comprised of:
    - attributes (values in a 1-1 relationship with the record)
    - tables (values in many-1 relationship with the record). (These are *not database tables*, but abstract tables)

- **Attribute queries**: SQL queries that supply attribute values to records

- **Table queries**: SQL queries that supply table values to records

- **Questions**: abstract representations of searches. These include, for example, help text and result column configuration. They reference an underlying query.

- **ID queries**: the underlying queries that implement questions. These may be either SQL or web services. They return a list of record IDs.

- **Parameters**: the parameters users can supply for individual ID queries. The three types are:
    - string
    - enum
    - vocabulary

- **Vocabulary queries**: the SQL that provides values for vocabulary parameters

The View

The WDK view creates a website that displays the model. It uses Java Server Pages (JSP). The JSP pages have access to the searches and records in the model and display them. The view also defines custom JSP tags that help create JSP pages.

A default view of the model is provided. After you set up Tomcat, install the WDK and define a model, you will have a working web site similar to the template site. As described below, you then configure it to give a "branded" look.

The Controller

The WDK controller functions internally to the WDK. It conforms to standard Model 2 architecture, and uses the Struts framework.

Strategies WDK: Requirements
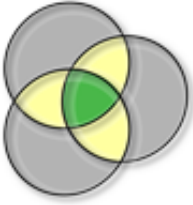
Strategies WDK Documentation

WDK Version: 2.0 (4/29/2011)

Please see the Strategies WDK Project Home for details about the project and useful links, etc.

Chapters
1. Overview
2. **Requirements**
3. Installation
4. Creating a site
    a. Designing a model
    b. Configuring and customizing the view

2. Requirements

Software and hardware
Database schema

Software and hardware

Operating system	RedHat Enterprise Linux 4
	RedHat Enterprise Linux 5
	CentOS 5.5
	CentOS 5.6
	Windows - not compatible
Java	1.5+
Web server software	not required
Web server hardware	4GB RAM, single Xeon E5620
Web application software	Apache Tomcat 5.5
	Apache Tomcat 6.x
Database server RDBMS	Oracle Enterprise Edition 10g

	Oracle Enterprise Edition 11g
	PostgreSQL server 8.x+
Database server hardware	16GB RAM, dual Xeon X5650 variable depending on needs
Browser	Firefox 3+, Chrome, Safari are fully supported; IE 8+ are partially supported.

Database schema

- Entities that will be represented in the WDK as records must have either a one-part, two-part or three-part *unique* primary key where both columns of the key must be of type STRING.
- It is recommended to choose stable ids as the primary key for your record types, since the id will be exposed in the URL.
- All attributes of all records must be available to the WDK engine through SQL queries against the RDBMS. An exception to this rule is the dynamic attributes returned by Web Service based processQueries. (See *Attribute queries* in the Model section, and for details, see Designing a model)


Published by Google Drive – Report Abuse – Updated automatically every 5 minutes

---

Strategies WDK: Installation

Strategies WDK Documentation

WDK Version: 2.0 (4/29/2011)

Please see the <u>Strategies WDK Project Home</u> for details about the project and useful links, etc.

Chapters

3. Installation

Installing the WDK is a multi-step process. Expect it to take on the order of an hour to complete. After it is installed you will use it to create your site, design your model and configure and customize your view. These tasks are the bulk of the work in deploying the WDK and may take significant developer time. (After reading the documentation for those steps, if you have additional questions about what will be involved for your application, drop a line to the WDK Users mailing list)

Understanding the install targets

There are two targets of the installation
- `$GUS_HOME` - This is the location that you install the model to when testing it.
- The webapp in the web application server - This is the location you install the whole WDK when it will run as a website

To get started, you will install and test the WDK model. You can set your `$GUS_HOME` to be anywhere, such as a `gus_home` directory in your home directory. You will be able to design your own model against the data in your database. You will do significant development and testing of your model without running in the trickier context of a web application.

After you are satisfied with your model, you will install into the web application server. To do so, set up a `GUS_HOME` that is in the same file system as your web application directories. (You will be creating symbolic links from the web application directory into `GUS_HOME`.) Then follow the instructions below for installing the webapp.

Using the GUS install system

The WDK uses the GUS install system, which is included with the WDK releases. Follow the [GUS Installer instructions](#) to set it up.

Downloading

Get the latest release of the WDK from [here](#)
Unpack it into the `$PROJECT_HOME` you set up for the GUS install system.
Then, install the WDK into `$GUS_HOME`:

```
        build WDK install -append
```

Installing the database driver

The current release supports Oracle and PostgreSQL.

The PostgreSQL JDBC driver is included in the distribution.

For licensing reasons, *the Oracle JDBC driver is not included in the distribution*. To include it:
- get a copy from your system administator or visit the [Oracle downloads site](#)
- copy it into `$GUS_HOME/lib/java/db_driver`
- (the build system provides an "IMPORTANT REMINDER" to alert you to this)

Installing the TemplateDB Website

Installing the template model

The WDK release includes a TemplateDB Website. You will need to install and play with it before you are ready to build your own site. The Template Site has a model and a view (just like yours will).

Every WDK driven project has a project_id, and the project_id of TemplateDB Website is `TemplateDB`.

WDK uses the project_id in many places, for example, WDK will try to locate the configuration file under a directory with the same name as project_id, in the `$GUS_HOME/config/`. Also, you must provide project_id as the input argument for all WDK commands, such as:

```
wdkCache -model TemplateDB -reset
```

After you have installed the WDK into `$GUS_HOME`, there are three additional steps to installing the Template model. You must configure the model, create a query cache and create the Template database.

Understanding the cache

Before going on to configure the model and then create the query cache, it will help if you understand the purpose of the cache.

The WDK model stores query results in a disk "cache." The cache is in your database. (In the section #Configuring the model you will configure the model to tell it where to create the cache tables.)   As you will see below, each search has an underlying query of some type (sql or web service). If the same query is requested again with the exact same parameter values (regardless of which user requests it), the result is fetched from the cache, avoiding the expense of running the query all over again.

When you define your model (see #The Model XML File), you will designate which queries to cache and not to cache, based on your expectations of your system's use. By default, queries that are used in Questions are cached, and you can turn that off by setting the isCacheable attribute to false. Queries that are used in Records are not cached.

The WDK is delivered with the Jakarta Commons DBCP connection pooler.

NOTE: Whenever you change the definition in your model of a cached query then *you must reset the cache (see #Creating and managing the cache)*.

Configuring the Template model

To configure the Template model, you must edit its configuration file:
`$GUS_HOME/config/TemplateDB/model-config.xml`

Here is an example of the model-config.xml:

```
<?xml version="1.0" encoding="UTF-8" ?>
<modelConfig modelName="templateModel"
    webServiceUrl="http://your_website/webapp/services/WsfService"
    smtpServer="127.0.0.1"
    supportEmail="support@your_website"
    defaultRole="wdk_user"
    secretKeyFile="/your_wdk_secret_key"
    adminEmail="admin@your_website"                     >

    <appDb login="your_app_db_login"
        password="your_app_db_password"
        connectionUrl="your_app_db_connection"
```

```
        platform="PostgreSQL"
        userDbLink="" />

    <userDb login="your_user_db_login"
        password="your_user_db_password"
        connectionUrl="your_user_db_connection"
        platform="PostgreSQL"
        userSchema="wdkuser."
        wdkEngineSchema="wdkengine." />

    <emailSubject>
        <![CDATA[ Your TemplateDB Password ]]>
    </emailSubject>
    <emailContent>
        <![CDATA[ The content of the email user will get when they register or
reset password on your site.]]>
    </emailContent>
</modelConfig>
```

The properties and values you can set in the <modelConfig> are:

modelName	required	value: the name of the top level model xml file (no .xml extension). WDK will look for the model file at $GUS_HOME/lib/wdk/<modelName>.xml.
webServiceUrl	required	value: the URL to the WSF service that will be used by the processQueries.
smtpServer	required	value: the IP of an SMTP server. WDK uses it to send out emails when user register or change password.
supportEmail	required	value: the reply address WDK uses in users' registration/change-password emails
secretKeyFile	optional	value: the absolute path to a file that contains secretKey. WDK will will use this key to encrypt user's cookie on the client machine.
defaultRole	optional	value: user's role name. default: wdk_user WDK will assign the role to newly registered users.
adminEmail	optional	value: admin email. If it is present, when an exception occurs on the website, WDK will try to send an email to this address with details about the exception.
useWeights	optional	value: true/false default: true enables the weighting feature on queries.
<appDb>	required	see <appDb> section

`<userDb>`	required	see <userDb> section
`<emailSubject>`	required	the subject of user registration/change-password email.
`<emailContent>`	required	the content of user registration/change-password email. Three macros are supported in the content: $$USER_NAME$$: user's display name, which is a combination of first name and last name; $$EMAIL$$: user's registration email; $$Password$$: the system generated temporary password for the user.
`<queryMonitor>`	optional	see <queryMonitor> section
`<paramRegex>`	optional	overrides the default regex used to validate the input values of all stringParams

The properties and values you can set in <appDb> are:

`login`	required	the login name to the application database.
`password`	required	the password for the login name
`connectionUrl`	required	the JDBC connection string to the application database. The queries defined in the model file will be executed in this database.
`platform`	required	value: Oracle/PostgreSQL WDK currently supports only Oracle and PostgreSQL
`userDbLink`	required	If your user database is different from application database, you need to provide a DB LINK from application database to the user database; the name should starts with "@". If don't use db link, please leave it as an empty string.
`maxActive`	optional	value: positive integer, default: 20 Connection pool setting, the maximium number of active connections allowed.
`maxIdle`	optional	value: non-negative integer, default: 1 Connection pool setting, the maximium number of idle connections allowed
`minIdle`	optional	value: non-negative integer, default: 0 Connection pool setting, the minimium number of idle connections allowed.
`maxWait`	optional	value: non-negative integer, default: 50 Connection pool setting, the maximium number of waiting connections allowed.

showConnections	optional	value: true/false, default: false print out the connection usage periodically, it is useful to debug connection leaks. This option is required if showConnectionsInterval or showConnectionsDuration is present.
showConnectionsInterval	optional	value: positive long integer, default: 10 the interval, in seconds, between each output of connection count.
showConnectionsDuration	optional	value: positive long integer, default: 600 How long the showConnection utility will run, when the duration is reached, the utility will stop printing connection usage.

The properties and values you can set in the <userDb> are:

login	required	the login name to the user database.
password	required	the password for the login name
connectionUrl	required	the JDBC connection string to the user database. The user related tables are created in this database.
platform	required	value: Oracle/PostgreSQL WDK currently supports only Oracle and PostgreSQL
userSchema	required	The schema used to create user dependent tables, if you use a schema other than the default schema, you need to specify it here, ending with ".".
wdkEngineSchema	required	The schema used to create user independent tables, if you use a schema other than the default schema, you need to specify it here, ending with ".".
maxActive	optional	value: positive integer, default: 20 Connection pool setting, the maximium number of active connections allowed.
maxIdle	optional	value: non-negative integer, default: 1 Connection pool setting, the maximium number of idle connections allowed
minIdle	optional	value: non-negative integer, default: 0 Connection pool setting, the minimium number of idle connections allowed.
maxWait	optional	value: non-negative integer, default: 50 Connection pool setting, the maximium number of waiting connections allowed.
showConnections	optional	value: true/false, default: false

		print out the connection usage periodically, it is useful to debug connection leaks. This option is required if showConnectionsInterval or showConnectionsDuration is present.
`showConnectionsInterval`	optional	value: positive long integer, default: 10 the interval, in seconds, between each output of connection count.
`showConnectionsDuration`	optional	value: positive long integer, default: 600 How long the showConnection utility will run, when the duration is reached, the utility will stop printing connection usage.

The properties and values you can set in the <queryMonitor> are:

`slowQueryThreshold`	optional	value: non-negative long integer, default 10 the running time threshold, in second, for a sql to be considered as a slow query. the query will be reported in the log.
`brokenQueryThreshold`	optional	value: non-negative long integer, default 60 the running time threshold, in second, for a sql to be considered as a broken query. the query will be reported in the log, and an email about this query will be sent to the admin. (the admin's email is configured in <modelConfig> section.)
`<ignoreSlowQueryRegex>`	optional	If a regex is specified, any slow sql that contains the pattern will be ignored, and not logged.
`<ignoreBrokenQueryRegex>`	optional	If a regex is specified, any broken sql that contains the pattern will be ignored, and not logged, no email will be sent out for that sql.

Creating and managing the cache

After you have edited the model config file, create an empty cache by using the wdkCache command. Here is its usage:

```
% wdkCache
usage: wdkCache -model model_name -new|-recreate|-reset|-drop

Create, reset or drop WDK query cache. (the cache tables are placed in the
schema owned by login).


Options:
-model <project_id>                    the project id. This is used to find the config file
         ($GUS_HOME/config/<project_id>/model-config.xml).
-drop              drop all cache tables and cache index tables.
-new               create cache index tables used by WDK.
```

`-reset`   drop all cache tables, but doesn't drop cache index table and, don't reset cache related sequences.

`-recreate`   drop all cache tables and cache index tables, then recreate all the index tables.

`-forceDrop`   optional, need to be used together with `-drop`, `-reset`, or `-recreate`, drop dangling cache tables that was not referred in the index table (created due to some error).

`-dropSingle <query_instance_id|query_full_name>`   If an instance id is given, WDK will only remove the cache for the specified query instance; if a query's full name is given, WDK will remove the cache for all instances of the query.

`-show`   display the cache usage statistics.

To set up the Template model's cache use this command:
```
% wdkCache -model TemplateDB -new
```

Setting up the Template database

The WDK release includes a Template database. It is packaged in a set of files in $GUS_HOME/data/WDK/TemplateModel/testTables. The model regression test makes use of the Template database (as does the Template website). When you run the regression test it optionally creates the Template database in your RDBMS, moving the data from the files into tables. If you want to manage the Template database yourself, use the wdkTestDb command:
```
% wdkTestDb

Create a Template database to use in testing the WDK. (The database is created
from files included in the WDK distribution.)

usage: wdkTestDb -model model_name [-create | -drop]

Options:
-model <model> The name of the model. This is used to find the Model config
file ($GUS_HOME/config/model_name-config.xml)
```
Use this command to create the Template database:
```
% wdkTestDb -model TemplateDB -create
```

Running the Template Model sanity test

The Template Model has a companion set of sanity test. Running it is a good way to exercise the WDK code installed.
```
usage: wdkSanityTest -model model_name -verbose

Run a test on all queries and records in a wdk model, using a provided sanity
model, to ensure that the course of development hasn't dramatically affected
wdk functionality.
Options:
-model <project_id>                       the project id. This is used to find the config file
        ($GUS_HOME/config/<project_id>/model-config.xml).
-verbose          Print out more information while running test.
```
Use this command to test the Template model:
```
% wdkSanityTest -model TemplateDB
```

Running the Template Model regression test

The regression test runs a set of wdk commands, and compares the results to a previously run, manually-validated expected result. If the output matches, the test passes, otherwise it fails. The test ensures that expected behavior of the codebase remains consistent even when behind-the-scenes changes are made. The WDK release provides a standard regression test to use with the Template model; if this test fails when run after installing the Template model, something is seriously wrong. The test can be found at

```
$GUS_HOME/data/WDKTemplateSite/Model/regressionTest/sampleRegressionTestCommands
```

The test uses data in the Template database. Set the --commandListFile flag to point to this file when running the wdkRegressionTest command:

```
usage: Runs wdk executables from provided file and compares output to an
expected result for testing purposes

wdkRegressionTest
--configFile (database configuration file to use with wdk model)
--loadNewDatabase! (flag to create database from provided test flat files; set
this flag
when running the regression test for the first time and when data in the files
have changed)
--createNewExpectedResult! (overwrite existing regression test expected result)
--outputDirectory (all results of test including error files will be placed in
this directory)
--commandListFile (file that contains lists of commands to run in test)
--verbose (prints out names of shell commands being executed)
```

In typical usage, you should run the regression test provided with the WDK release. If you find a reason to create your own regression test, then create a separate command list file. The command list file includes one or more tests. Each test is composed of three lines, and the next test, if any, follows beginning on the fourth line. The expected input for a test is as follows:

name="[name of test]"
command="[command to run with parameters conforming to the normal usage of the command, EXCEPT any global parameters (configFile) passed to the wdkRegressionTest commmand]"
critical="[true or false; if true, the regression test will immediately exit on failure, without running any more of the provided tests]"

Installing the Template View

Configuring Tomcat

The WDK uses a slightly unconventional installation strategy. It installs a complete and working GUS_HOME in the same file system as the web application directory, and uses symbolic links to link into it from the web application directory. The reason is that this provides a working GUS_HOME so that you can test your model using the extensive command line tools available for the model. Because it is linked in, you know for sure that the model you are testing is the same model that is running on your site.
The complication with this strategy is that it requires you to configure Tomcat to allow symlinks. This is not a default configuration because Tomcat feels that it may open a security issue. We believe that as long as

nobody on your team makes symlinks to bad places, this is not a security problem. However, if you do not want to use links, then replace them with direct copies.
To configure Tomcat to allow symlinks

- do not use symlinks that are cross file system boundaries
- you may not use Tomcat's optional security mode
- in Tomcat's $TOMCAT_HOME/conf/server.xml file, the <Context path=/> directive for the web application must have allowLinking="true" like this (substituting values specific to your site):

```
<host ...>
<Context path="/wdkTemplatesite"
        docBase="/www/wdkTemplatesite/webapp"
        debug="1"
        privileged="false"
        allowLinking="true" >
        <Logger className="org.apache.catalina.logger.~FileLogger"
            prefix="wdkTemplatesite-log."
            suffix=".txt"
            timestamp="true"
            docBase="/www/wdkTemplatesite/logs/" />
</Context>
```

Building the WDK on your site

Here are the steps required

- if you do not already have a GUS_HOME in the same file system as your web application directory, create an empty directory named, say, gus_home.
- set the GUS_HOME environment variable to that directory
- set up a "webapp" directory for your web application where tomcat will find it. You must also create a WEB-INF directory within your webapp directory.
- create a property file your_web_prop_file with the following information

```
webappTargetDir=your_webapp_directory
```

- run the build command

```
build WDKTemplateSite webinstall -append -webPropFile your_web_prop_file
```

- if you haven't already done so, copy your oracle JDBC .jar file into $GUS_HOME/lib/java/db_driver
- copy the oracle or postgres JDBC driver file (depends on which on you use) to your $TOMCAT_HOME/common/lib if it does not already contain the driver file

Testing the Template website

The WDK is now installed on your site. Bring up the TemplateDB WDK site to test it.
Here are the steps

- if you haven't already done so, edit $GUS_HOME/config/TemplateDB/model-config.xml to reflect your site.
- set up the header and footer files and the web.xml file

```
cd your_webapp_directory
cd WEB-INF
cp web.xml.Template web.xml
cd tags/site
cp footer.tag.Template footer.tag
cp header.tag.Template header.tag
```

- edit your_webapp_directory/WEB-INF/web.xml to provide a correct logging directory
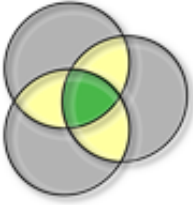- work with your system administrator to start up the application in Tomcat

---

Strategies WDK: Create a site

Strategies WDK Documentation

WDK Version: 2.0 (4/29/2011)

Please see the [Strategies WDK Project Home](#) for details about the project and useful links, etc.

Chapters

---

4. Creating a site

Creating your project

Now that you have installed and tested the Template Site you are ready to create your own WDK-based project. The first step is to set up a directory structure that is compatible with CVS and the GUS installer. This way you will be able to keep your project safe and also use the GUS installer to install your project along with the WDK.

Using WDKTemplateSite as a template

To get started use the WDKTemplateSite as a template. You will need to chose the name of your site (referred to below as [TheNameOfMySite](#) ?). You will also need to chose the name of your model, (referred to below as myModelName).
Start out by doing this:

```
% cd $PROJECT_HOME
```

```
# init your project with WDKTemplateSite
% cp -r WDKTemplateSite TheNameOfMySite
% cd TheNameOfMySite/Model
# if your copy of the WDK came from CVS, remove the CVS dirs
% rm -r `find . -name CVS`
# remove unneeded data/ dir
% rm -r data
% cd config
# rename config files so they use your model name
% mv TemplateModel.xml myModelName.xml
% mv TemplateModel.prop myModelName.prop
% mv TemplateModel-config.xml myModelName-config.xml
% mv TemplateModel-sanity.xml myModelName-sanity.xml
# remove unneeded regression test file
% rm regressionTestModelConfig.xml
% cd $PROJECT_HOME/TheNameOfMySite/Site/webapp/WEB-INF
# move the Template's web.xml file to a template file for your project
% mv web.xml.Template web.xml.template
% cd tags/site
# move the Template's header and footer into files for your project
% mv header.tag.Template header.tag
```
% mv footer.tag.Template footer.tag

Editing the template

Now your project's directory structure is pretty close to what you need. To finish it off, you need to:
- modify myModelName-config.xml as described above in #Configuring the model
- re-write myModelName.xml and myModelName.prop as described below in #Creating a model
- re-write myModelName-sanity.xml as described below in #Creating a sanity test
- configure your view as described below in #Configuring the view
- edit the file $PROJECT_HOME/TheNameOfMySite /build.xml and replace "WDKTemplateSite" with TheNameOfMySite. This will enable you to install your project using the GUS installer

Building your project

To build your project, follow the directions in either #Downloading (to install the Model alone without the view) or in #Building the WDK on your site to install the project into a web site. In both cases, substitute TheNameOfMySite for WDKTemplateSite in the build command to build *your* project instead of the Template Site.

Storing your project in CVS

It is recommended that you store your project in CVS. To do so, do this:
```
% cd $PROJECT_HOME
% mv TheNameOfMySite TheNameOfMySite.bak
```

```
% cd TheNameOfMySite.bak
% cvs -d TheNameOfMyRepository import -m "Start of project" TheNameOfMySite
dontcare start
% cd ..
% cvs -d TheNameOfMyRepository co TheNameOfMySite
```

Published by **Google Drive** – **Report Abuse** – Updated automatically every 5 minutes

---

EuPathDB catalog of hardware and required software.

Server and Network Infrastructure
The server infrastructure is hosted in institutionally-owned datacenters that provide redundant electrical power, fire suppression and ambient air cooling. The hosting institutions also provide and maintain the network infrastructure that connects EuPathDB hardware to the public wide-area network. Similarly, the compute clusters required for large scale data analysis is property of the hosting institutions.

The hardware architecture owned by and transferable from EuPathDB consists of the following:

Oracle database and administrative Servers
   5 X Dell PowerEdge R610

Production webservers
   1 X IBM x3550
   1 X Dell PowerEdge R610

Developement webservers
   1 X Dell PowerEdge R610
   1 X IBM x3550

Software integration server
   1 X Dell PowerEdge R610

Data processing servers
   1 X IBM x3550

Backup server
   1 X Dell PowerVault DL2000

Virtual machine host
   1 X Dell PowerEdge R610

SAN Storage
   1 X Arch SS10A Storage System
   5 X EqualLogic SAN arrays

Infrastructure administration
   1 X Dell PowerEdge R310

Network switches
   3 X Brocade 48 port switches

All servers run CentOS 5.8. The databases are hosted in Oracle 11.2.0.3. Webservers require Apache HTTP server and Tomcat Application Server, both freely available, open-source applications. Backups at the University of Georgia are managed with Commvault, Amanda is used at the University of Pennsylvania. The servers and SAN storage are interconnected through a private VLAN on the Brocade switches. The switches are connected to the institutional public network on a separate VLAN.

Third-party Data Analysis Software

The following is a non-exhaustive list of third-party software used for data analysis. Many have licensing restrictions that preclude EuPathDB from transferring directly. Most have additional software or library dependencies for installation and function. These dependencies may not be included below but will be listed in each software's respective documentation.

Ant
Bioconductor for R
blat
bowtie
bowtie2
bwa
clustalw
cndsrc
consed
exportpred
GATK
HMMER
java
kohgpi
mavid
MCL
mdust
MUMmer
ncbi-blast
phrap
phred
phylip
primer3
psipred
RepeatMasker
RMAExpress
samtools
SignalP
smartmontools
SRA Toolkit
stampy

TMHMM
tmpred
TRF
tRNAscan-SE
varscan
weblogo
wu-blast