Arcturus, Eimeria

And All That

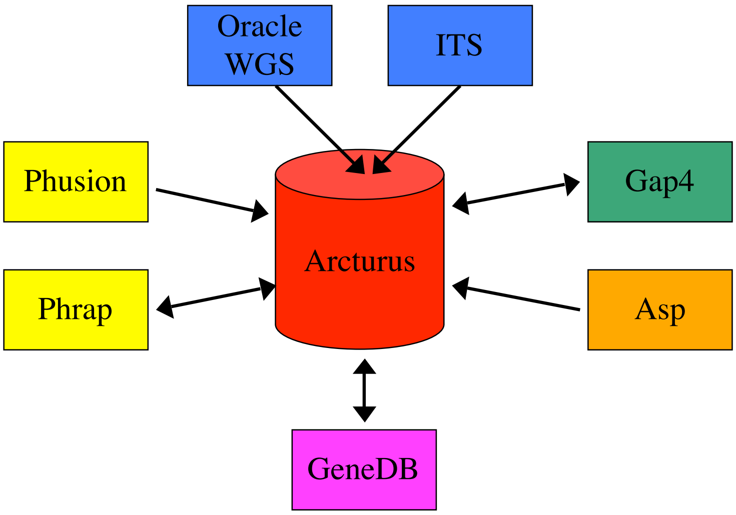
Ed Zuiderwijk

PSU Informatics

About Arcturus

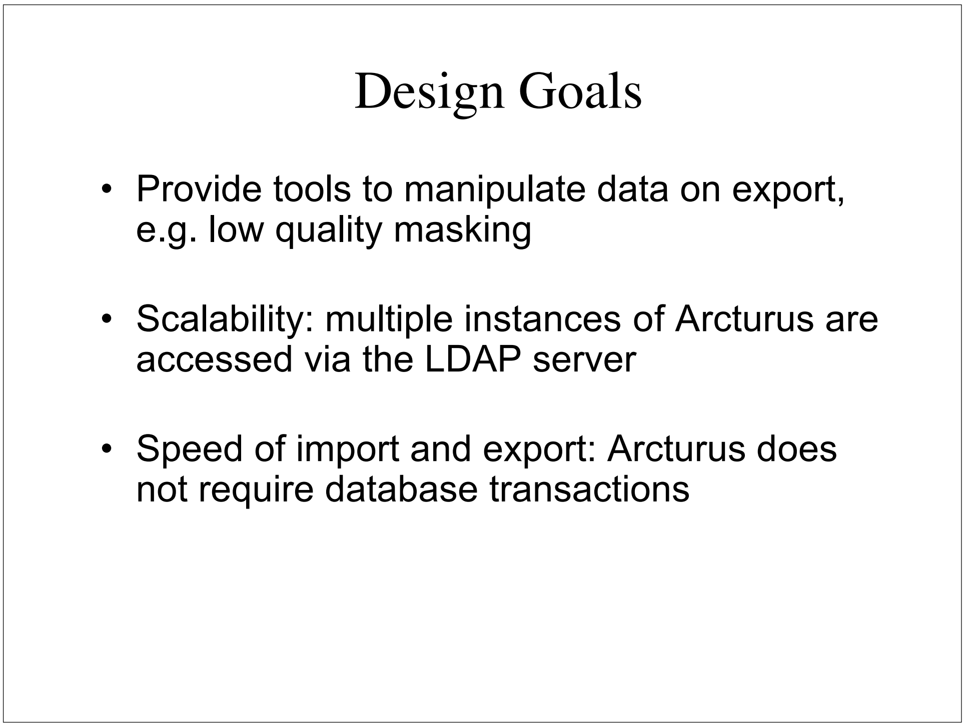
- An assembly-management system developed in the PSU to assist in the finishing process of large (WGS) pathogen genomes.
- Based on the MySQL database engine.
- Complex software system with a Perl back-end and a Java GUI front-end, developed using OO design.
- Has over the past year come into production and is now used to manage some of the PSU projects.

Arcturus Interactions



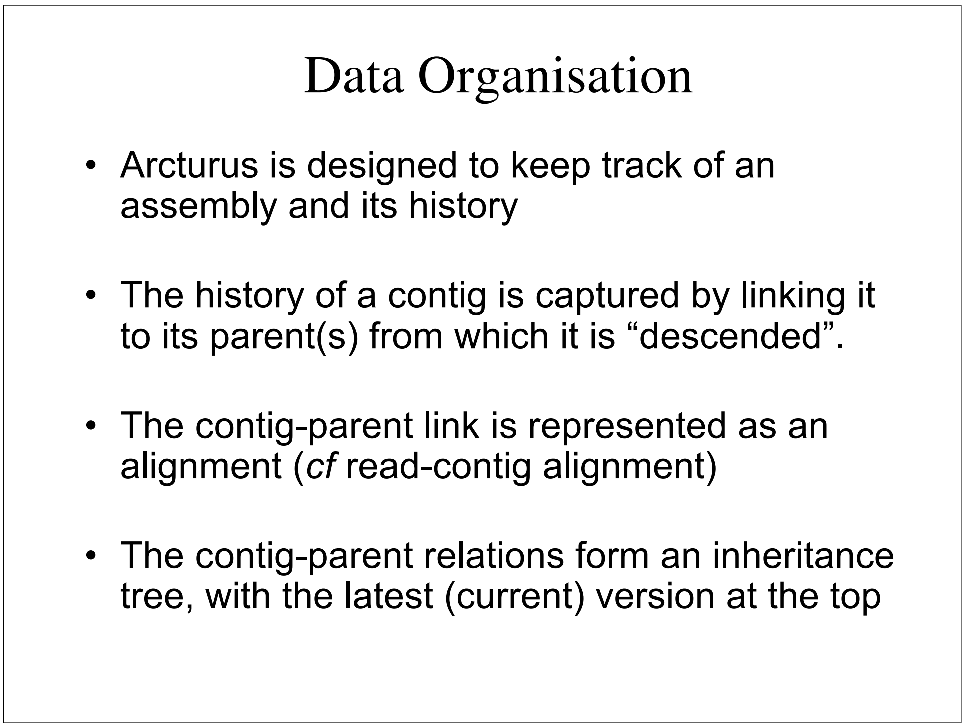
Design Goals

- Export assembly splits to Gap4 or into the assembly pipeline, and import results
- Provide mechanisms for defining splits, e.g. by finding scaffolds
- Provide mechanisms for moving contigs between splits
- Safeguard against multiple users accessing the same data



Design Goals

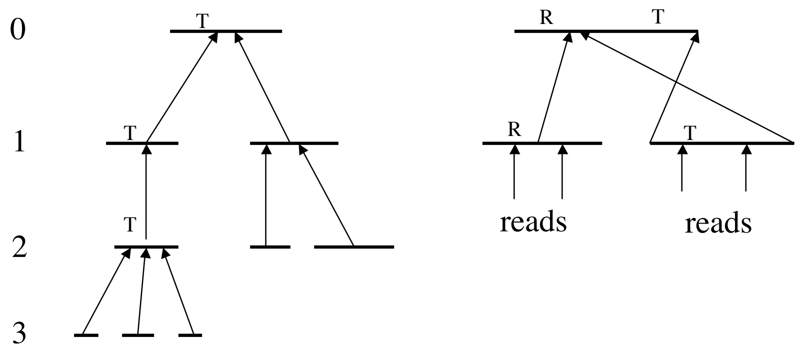
- Provide tools to manipulate data on export, e.g. low quality masking
- Scalability: multiple instances of Arcturus are accessed via the LDAP server
- Speed of import and export: Arcturus does not require database transactions



Data Organisation

- Arcturus is designed to keep track of an assembly and its history
- The history of a contig is captured by linking it to its parent(s) from which it is “descended”.
- The contig-parent link is represented as an alignment (*cf* read-contig alignment)
- The contig-parent relations form an inheritance tree, with the latest (current) version at the top

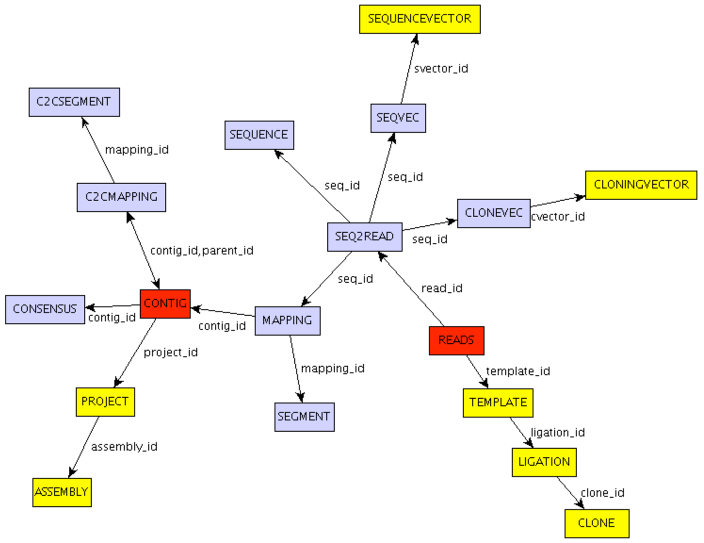
Inheritance Tree



Data Organisation

- The current generation of contigs consists of the ones which are not a parent themselves (found by a simple left join on two tables)
- On presentation of a new contig, the links to its parents (if any) are established and the new contig is added at the top of its tree.
- Arcturus recognizes if a contig is already present
- The alignment information is used to port annotation (tags) from one generation to the next.

Challenges



contig-to-contig alignment

- Alignment between a contig C and its parent contig P is based on analysis of read-to-contig alignments for the reads in common.

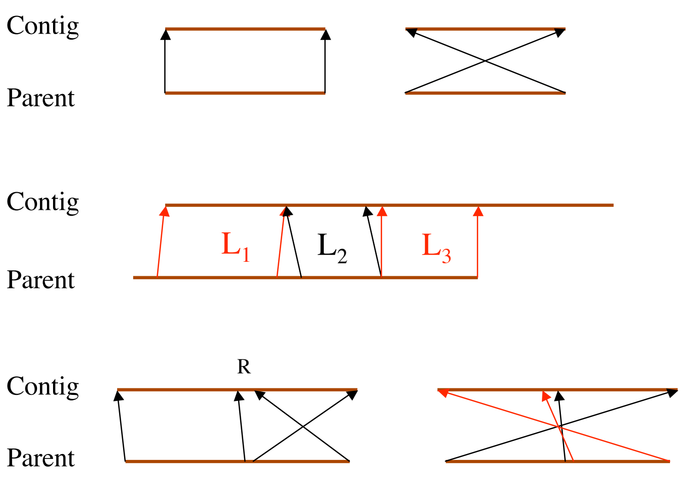
- The alignment of a read segment R to contig C is described by a linear operator L_C; *ibid* the alignment of R to P by L_P.

- The alignment between C and P at the position of the read is then given by the product operation:

  L_{CP} = (Inverse of L_C) * L_P

- Contiguous stretches on C and P having the same transformation parameters form the segments of contig-to-parent alignment.

Contig-to-parent Alignment Pathology



Application to Eimeria Tenella

- 60 Mbase WGS genome; 21000 contigs
- Many repeats; assembly proved difficult to handle by existing assembly tools, i.p. integrating Gap4 work by the finishers was cumbersome (if not impossible).
- By using Arcturus we were able to integrate finishing reads into the assembly, determine scaffolds and joins. We reallocated a large number of contigs to the various finishing projects (splits). The total number has now been reduced to about 3000 (in the splits) which cover 80% of the genome.

Applications

- Leishmania Braziliensis (33 Mbase)

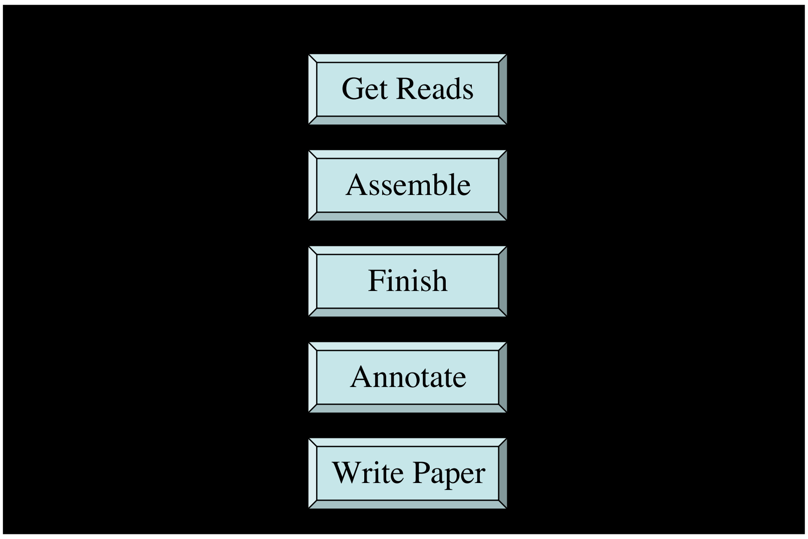
The original allocation to 36 splits was done by synteny with L.major. Using Arcturus' scaffold-finding tools we found that this allocation was not correct for about 5% of the sequence. The re-organized splits were exported to create new Gap databases.

The whole process took about an hour: exporting all 36 splits took less than 5 minutes by organizing the process in 36 batch jobs (of which 16 ran in parallel).

New Directions

- Automation of the contig-to-split allocation process by using a queueing system.
- Inclusion of FPC maps
- Reporting tools

ArcturusXP (v.11)



Get Reads

Assemble

Finish

Annotate

Write Paper

Thanks Folks

David Harper

Karen Mungall
Carol Churcher
David Harris
Barbara Harris
&
PSU Finishing Teams

Marie-Adèle Rajandream

Rob Davies
James Bonfield
Zemin Ning