SMRT Analysis v2.3 Software Release

October, 2014
SMRT® Analysis v2.3

- Iso-Seq™ (cDNA) Analysis
  - Updates to User Interface
  - Expanded algorithm parameters
- Long Amplicon Analysis
  - Support for mixed amplicon sizes
  - Support for longer amplicons (e.g. HLA class II)
- Performance Enhancements
  - Demultiplexing
  - Resequencing
  - cDNA Analysis
- Data and workflow standardization
  - BAM data format
SMRT® Analysis 2.3.0 enables exploration of diverse application areas through reliable, user-friendly workflows

<table>
<thead>
<tr>
<th>Application</th>
<th>Feature</th>
<th>User Benefit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Long Amplicon Analysis (HLA)</td>
<td>1. Long amplicons – e.g., HLA class II</td>
<td>Robust analysis for amplicon pools of mixed sizes - e.g. full-length HLA Class I &amp; II genes</td>
</tr>
<tr>
<td></td>
<td>2. Analysis of mixed amplicon sizes</td>
<td></td>
</tr>
<tr>
<td>Iso-Seq™ (cDNA) Analysis</td>
<td>SMRT Analysis UI for custom primers and algorithm parameters</td>
<td>Enhanced user experience, easy to learn SMRT Analysis, increased adoption rate</td>
</tr>
<tr>
<td>Robustness</td>
<td>SMRT Analysis environment and installation</td>
<td>Reliable analysis execution independent of customer compute environment, simple and easy to use installation process</td>
</tr>
<tr>
<td>Performance</td>
<td>1. Demultiplexing</td>
<td>Faster time-to results, efficient use of compute resources</td>
</tr>
<tr>
<td></td>
<td>2. Resequencing</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3. cDNA analysis</td>
<td></td>
</tr>
<tr>
<td>(BETA) Data and workflow</td>
<td>1. BAM data format</td>
<td>Increased adoption rate of support for PacBio data from 3rd party analysis tools, performance and scalability advancements</td>
</tr>
<tr>
<td>standardization</td>
<td>2. Code modularization and re-implementation</td>
<td></td>
</tr>
<tr>
<td>DevNet</td>
<td>1. Full Length Viral – Minor Variant – HIV Use Case</td>
<td>Continuous support for advanced analysis users through DevNet and Github</td>
</tr>
</tbody>
</table>
Iso-Seq™ Protocol for cDNA analysis – 2.3 enhancements

Iso-Seq Algorithm – Main Steps

- **Classify**
  - QC/Classify reads of insert into full-length or non-full-length, chimeric or non-chimeric reads – **2x-3x faster in 2.3**

- **Polish**
  - **ICE** - Iterative Clustering and Error Correction - predict *de novo* consensus isoforms using full-length non-chimeric reads – **8x less memory usage in 2.3**
  - **Polish** - assign non-full-length reads to the predicted consensus isoforms, and polish these isoforms using Quiver – **N*x faster in 2.3**
    *N is the number of subtasks
New in 2.3

• SMRT Portal User Interface for:
  – Custom primers
  – Full-length reads without polyA tails
  – ICE and Quiver parameters
    – Parallelize Cluster Polish jobs into multiple subtasks
    – Set accuracy criteria for categorizing consensus isoforms

• Protocol Setting
  – Parameters for Classify and Cluster are separated in the SMRT Portal UI

• Human readable annotations to predicted consensus isoforms (output format)
  – >c0/f18p78/3890
    isoform=c0;full_length_coverage=18;non_full_length_coverage=78;isoform_length=3890
Three Main Steps:

1. Coarse Clustering:
   - Groups reads from different amplicons into different clusters.
   - Detects read-read overlaps, constructing an overlap graph, then clusters the overlap graph to break the graph into the final clusters.

2. Phasing/Consensus:
   - The reads are loaded into the Quiver consensus calling framework, and an initial consensus of all reads is found.
   - ‘Mutation scores’ tested for the presence of SNPs that segregate the reads into multiple haplotypes.

3. Chimera Filtering
   - UCHIME algorithm detects chimeric sequences.
(Beta) Long Amplicon Analysis: New features in 2.3

• Long Amplicon Analysis:
  – Support for greater range of amplicon lengths (3-9 kb)
  – Enabled analysis of mixed sizes such as full-length HLA Class I and II genes
    – Can trim the ends off of consensus amplicons (e.g., primers)
    – Provide a fixed number of best-supported sequences (e.g., 2 for diploid)
      – Happens within fine-phasing analysis, so within each coarse-cluster
    – Can enable/disable coarse-clustering and/or fine-phasing analyses
      – Useful as a “barcoded consensus” algorithm

• Barcoding:
  – Filter reads based on minimum barcode score
  – Option to output separate FASTA files for each barcode

• Performance
  – Reduced memory usage (up to 40%)
Long Amplicon Analysis: SMRT® Portal UI enhancements

New in 2.3
SMRT Portal User interface for:

• Barcoding:
  • Minimum barcode score
  • Separate output files for each barcode

• LAA
  • Trimming the ends of consensus amplicones
  • enable/disable coarse-clustering and/or fine-phasing analyses
  • Provide a fixed number of best-supported sequences
SMRT® Analysis 2.3 Robustness Improvements

• Installation
  – Self-contained single tar ball includes all components needed
  – Automated installer goes to the process once launched
  – **New in 2.3** Enhanced command line usage for self-extracting installer/upgrader
  – ‘--help’ for top-level usage, ‘--helpall’ for all levels

• Analysis environment
  **New in 2.3** Enhanced stability of SMRT Analysis through separation from the user environment:
  – System and user-specified locales are ignored
    – Force to “C” (aka POSIX) default locale
    – Affects all code that runs under setup.sh, including SMRT Portal and all command line programs.
  – “setup.sh” related changes
    – setup.sh sets up environment for access to SMRT Analysis internals
    – Sourcing of setup.sh directly by command-line users is deprecated
    – setup.sh unsets almost all user environment variables
      – Except: USER, LOGNAME, PWD, TERM, TERMCPA, HOME, WORKSPACE, MPLCONFIGDIR, SMRT_*
    – New user-accessible bin directory (wrappers for SMRT Analysis internal programs)
      – $SMRT_ROOT/srmtcmds/bin
    – ‘smrtshell’ invokes a shell in the setup.sh context
**SMRT® Analysis 2.3 Performance Improvements**

- **New in 2.3** Faster resequencing analysis (**BETA** version)
  - Parallelizes merging and sorting routines over contigs – the major bottleneck for performance in all previous versions
  - Uses BAM as the alignment file format
  - (**BETA**) BAM_Resequencing_beta.1 protocol in SMRT Portal

---

**Old resequencing workflow**

- chunk_1.comp.h5 #pulsesloaded
- chunk_2.comp.h5 #pulsesloaded
- ...
- chunk_m.comp.h5 #pulsesloaded

  - CmpHtMerge
  - aligned_reads.comp.h5 #unsorted
  - CmpHtSort
  - aligned_reads.comp.h5 #sorted
  - h5repack
  - aligned_reads_contigchunk_1.bam #repacked
  - chunk by contig
  - quiver_contigchunk_1
  - variants_contigchunk_1.gff
  - variants.gff

**2.3 resequencing workflow**

- chunk_1.bam
- chunk_2.bam
- ...
- chunk_m.bam

  - samtools merge
  - samtools merge
  - ...
  - samtools merge
  - samtools sort
  - samtools sort
  - ...
  - samtools sort
  - aligned_reads_contigchunk_1.bam #repacked
  - quiver_contigchunk_1
  - variants_contigchunk_1.gff
  - variants_contigchunk_s.gff
  - variants.gff