High Performance Computing for High Content Screening Image Analysis – A Proof of Concept

S. Lee[1], F. Benmansour[2], M. Litherland[2], C. Fenoy[2], B. Holländer[1], F. Lauck[1], K. Palo[1], O. Deppe[1], J. Deihl[1], M. Daffertshofer[1]

(1) PerkinElmer, Inc. (2) Roche Pharma Research and Early Development, pREDi , Roche Innovation Center Basel

1 Overview

Here we report on a collaborative proof of concept study between F. Hoffmann-La Roche in Basel and PerkinElmer, Inc. where cluster-based high performance computing has been employed to significantly reduce the computing time for analyzing large amounts of high content screening image data.

Images captured by a PerkinElmer Opera™ QEHS system (151 384-well plates, three channels, nine fields per well) were extensively analyzed by an Acapella script that calculates approximately 300 descriptors per cell (nuclei, cytoplasm, membrane and spot segmentation; morphological, intensity and classification parameters). A high performance compute cluster based on Dell hardware, interconnected by InfiniBand for storage access and 10Gb Ethernet was set up to be controlled by a SLURM job scheduler, and each scheduled job analyzed a single well of a single microtiter plate.

Using this setup image analysis time was reduced by more than two orders of magnitude compared to an existing single server setup.

2 The Need for Faster HCS Image Analysis

The instrumentation to run effective screening campaigns in drug discovery is required to have as high a throughput as possible. Many choices have to be made to enable fast data acquisition in high content screening. To obtain statistically meaningful results large image fields capturing larger numbers of cells are needed without compromising the image resolution required to measure cellular phenotype with high accuracy. Larger fields can be imaged with cameras utilizing detectors measuring 2048 × 2048 pixels vs 1040 × 1400 pixels. The image acquisition time is optimized with high intensity light sources such as lasers and/or optimizing fluorescence staining resulting in reduced exposure times. If essential phenotypic features can be recognized with only two channels as opposed to three or more channels, large savings in screening times can be achieved.

Analyzing the data requires high performance computation to segment images with precision, and to quantify and classify cells by utilizing sophisticated algorithms. At the same time, the vast volume of image data challenges the performance of computing systems. Today, typical computing systems used in research laboratories for high content image analysis utilize multi-core SMP systems with ten or more cores and 4GB of RAM per core.

Figure 1 below lists typical image analysis times that can be achieved using PerkinElmer Columbus™ in the various stages of high content screening: Assay Development (Plate), Pre-Screen (Experiment), High Content Screen (Screen).

Data size: **20 GB** in 384 wells
First rounds of script development, QC and basic hypothesis validation

Computation time: **1-2 h**

6-10 plates: **120-200 GB**
Script optimization
Initial secondary analysis

Computation time: **6-20 h**

20-30 experiments: **2.4-6 TB**
Refinement of end-to-end analysis

Computation time: **5-17 days**

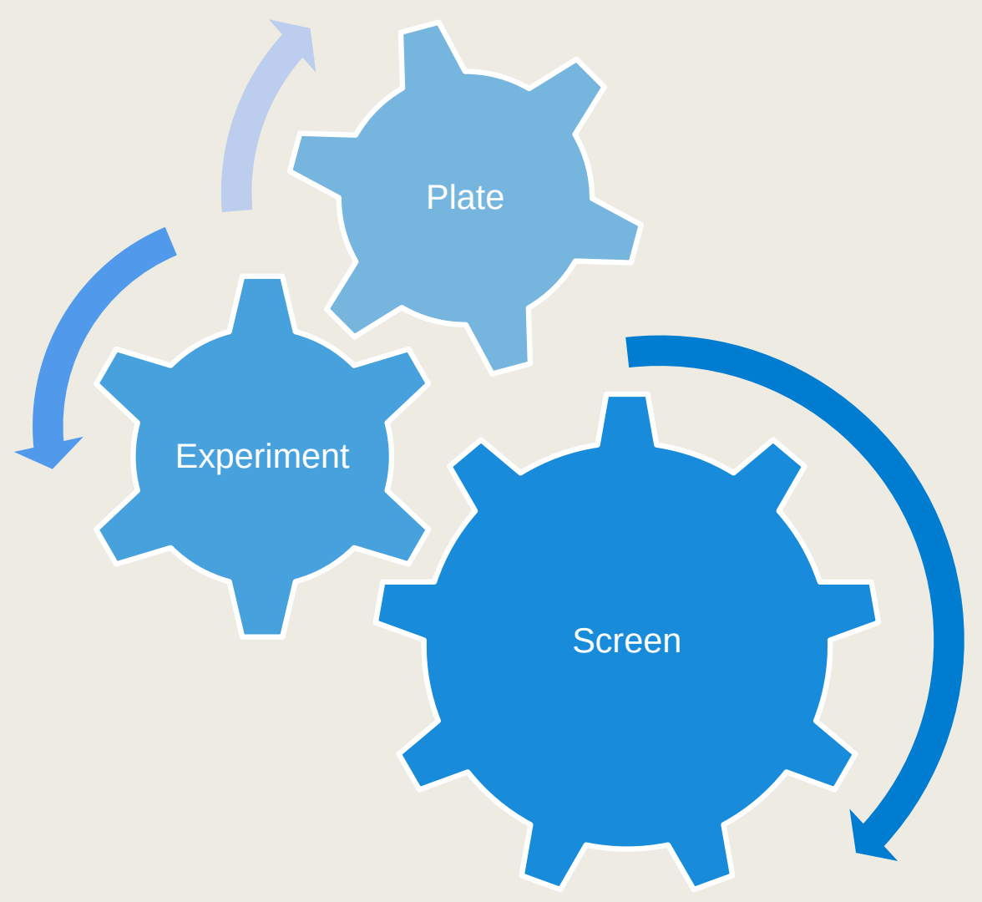


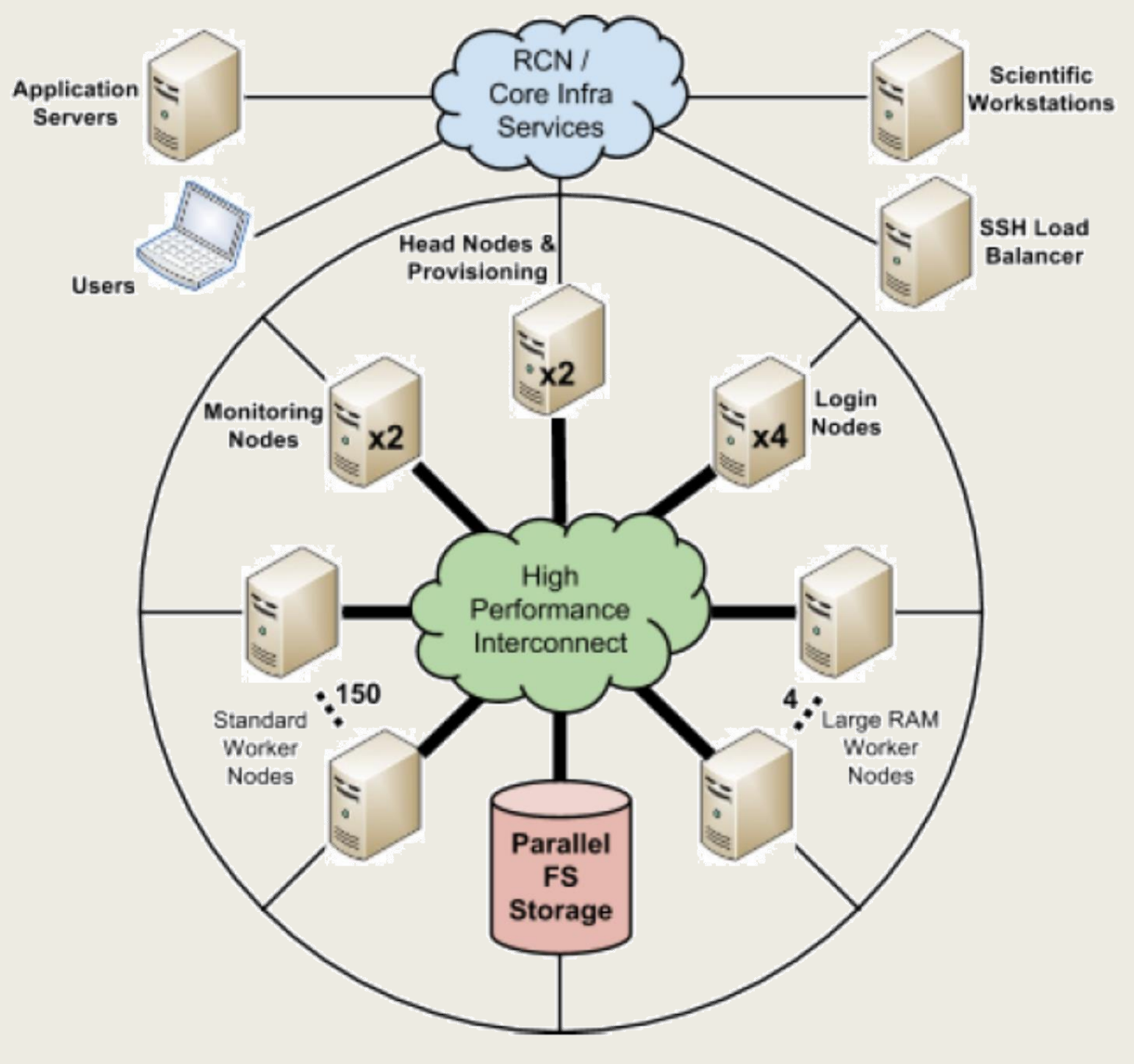
Figure 1: Typical computing times achieved with conventional systems for plates, experiments, and screens

Since batch analysis for full high content screens requires the most demanding computations, in this proof of concept, we have focused on exploring the limitation of high performance computational cluster systems to accelerate batch analysis for large screening campaigns.

3 HPC Hardware and Software Architecture

This proof of concept was performed on Roche’s corporate HPC environment: A cluster system consisting of several hundreds compute nodes based on Dell hardware, interconnected using InfiniBand for accessing the storage and 10Gb Ethernet. A parallel file system was used.

- Operating System: CentOS 7.2
- Job Scheduler: SLURM 15.08.4
- Application Management: Easybuild
- Image Analysis: Acapella 4.1



4 Data / Analysis Script

Images were captured with a PerkinElmer Opera QEHS and stored and analyzed for reference with PerkinElmer Columbus 2.7. The Analysis sequence was provided by an Acapella 4.1 script.

Image Data: 1331x985 pixels, 16bpp (three channels), nine fields per well
Screen Data: 384-well plates, 151 plates, > 3TB of image files

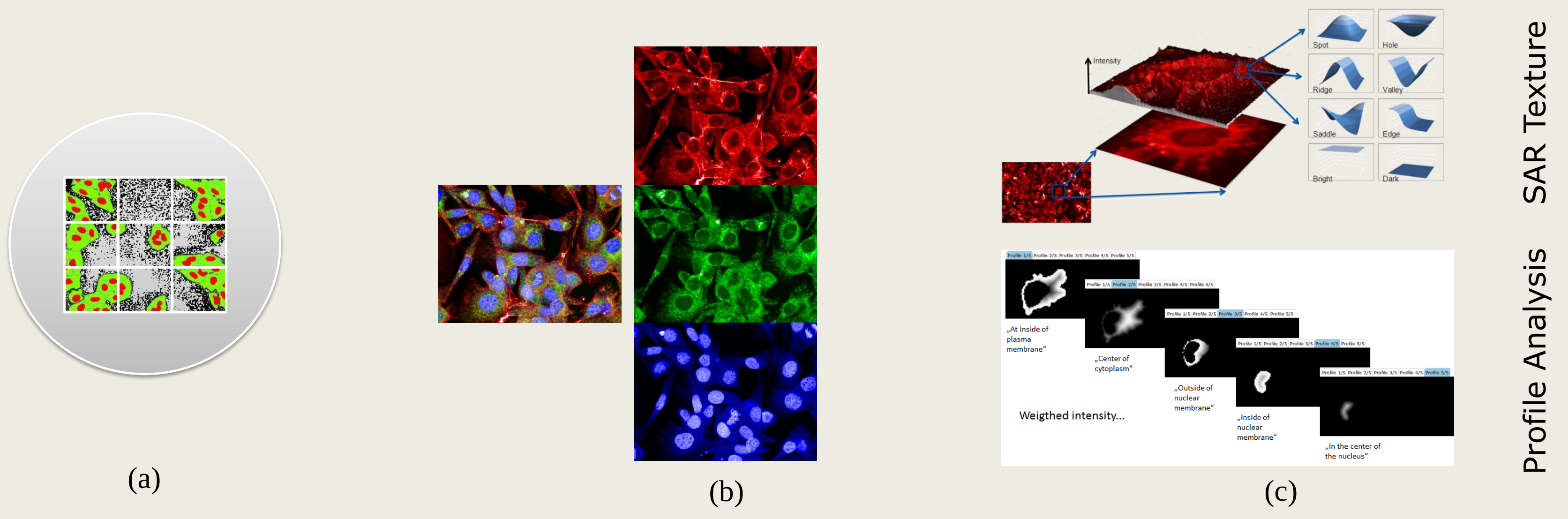


Figure 2: Examples: (a) nine fields per well, (b) three channels per field, and (c) texture and morphology analysis

5 Measurements

An Acapella Script was used to identify cells (nuclei, cytoplasm, and membrane) in the individual images. To describe the phenotypes, we calculated intensity, texture, and morphology parameters for each of the cellular compartments. Cluster analysis was used to classify cells according to their phenotype. In total, 300 different parameters were calculated.

As a baseline we performed a complete analysis using Columbus™ (12 cores, 4GB memory per core) and achieved an average compute time of 100 minutes per 384-well plate (305 and 308 measured wells). High performance compute jobs were run with a mximum of 1,800 cores per user as described in Section 3.

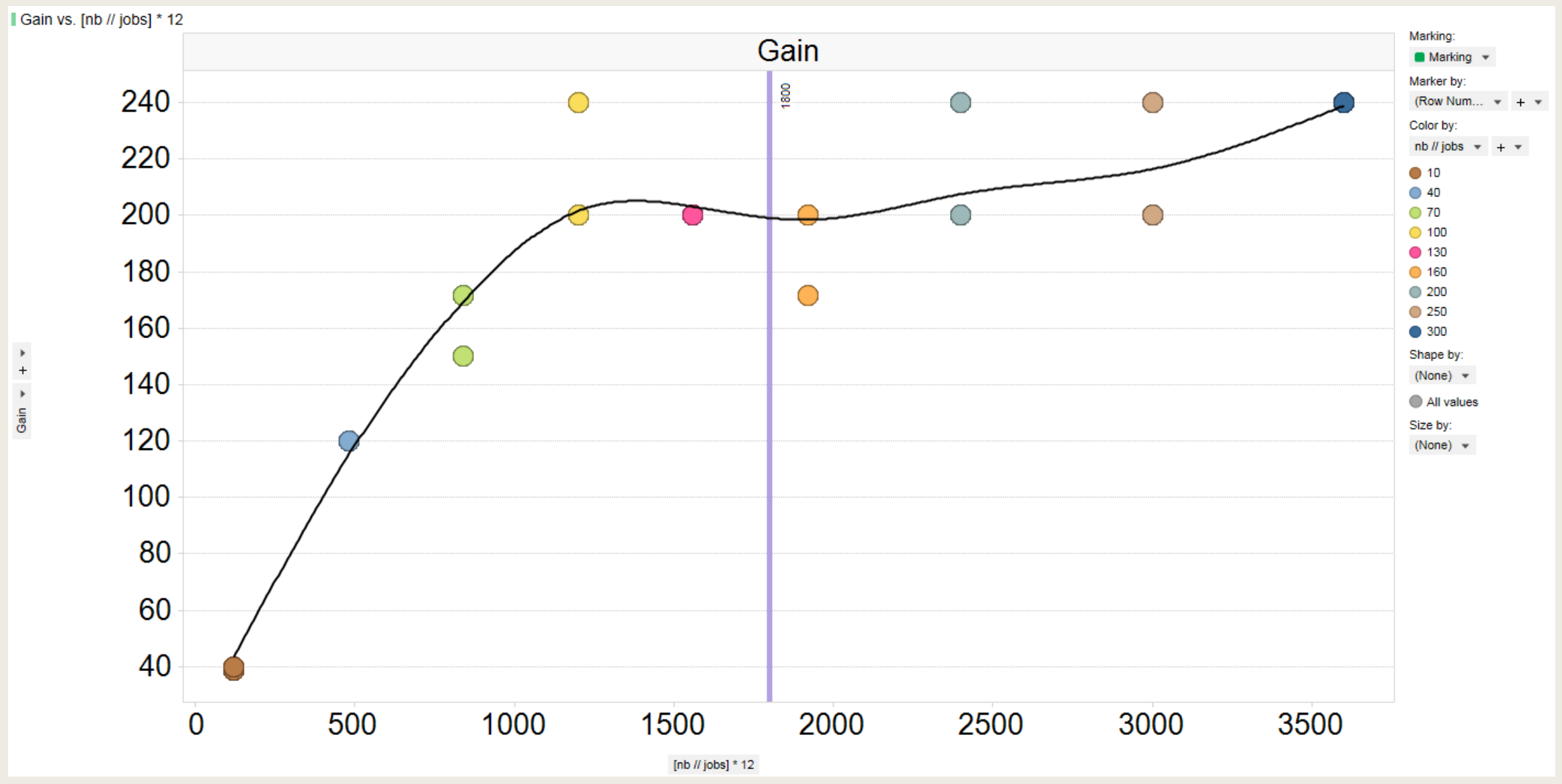


Figure 3: Each data point represents 12 microtiter plates. The x-axis indicates the number of (SLURM) requested cores per analysis run (# of cores per plate equals # of cores per run divided by 12). The y-axis indicates the *gain* – defined as the reference computation time (Columbus) divided by HPC computation time. Each run was repeated three times.

The vertical line marks the maximum of 1,800 cores per user.

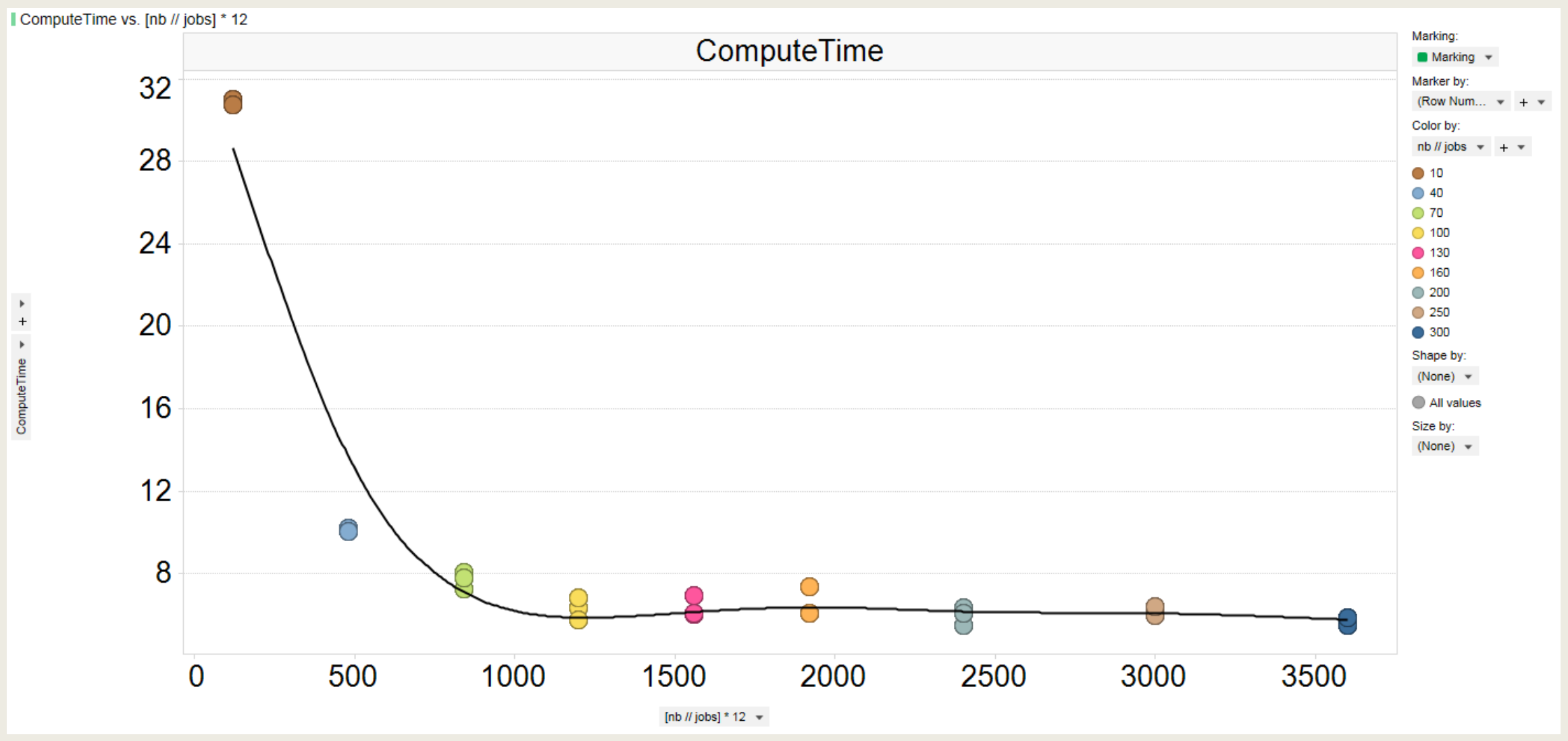


Figure 4: Each data point represents 12 microtiter plates. The x-axis indicates the number of (SLURM) requested cores per analysis run (# of cores per plate equals # of cores per run divided by 12). The y-axis indicates the HPC computation time. Each run was repeated 3 times.

Initially the gain achieved by well-based parallel computing scales almost linearly with the number of cores requested from the SLURM job scheduler. With less than 500 cores the gain already exceeds two orders of magnitude. Ultimately, the realized gains top out at 240 when the system approaches the limit of 1,800 cores per user.

6 Findings and Lessons Learned

Using Acapella on a HPC cluster can accelerate the speed of image analysis by at least two orders of magnitude. In the current proof of concept we observed a linear scale-up with the number of cores and the only limitation was set by the cluster policy; unexpectedly, we did not see any I/O-dependence on the overall performance. Even with a small number of cores (120) the computation time could be accelerated by a factor of 40 – reducing image analysis for a screen from two weeks to eight hours.

7 Summary

In a joint proof of concept we were able to demonstrate that high performance computing can accelerate High Content Image Analysis by two orders of magnitude or more. The performance gains will have a positive impact on high content screening campaigns allowing for better project planning and opening the doors for iterative analysis. Using high performance computing resources definitely accelerates HCS assay development and screening campaigns. PerkinElmer is looking at integration HPC support into future products.

PerkinElmer, Inc., 940 Winter Street, Waltham, MA USA (800) 762-4000 or +1 (203) 925-4602
www.perkinelmer.com