

2018

ISSN 1433-2620 > B 43362 >> 22. Jahrgang >>> www.digitalproduction.com

Publiziert von DETAIL Business Information GmbH

Deutschland
Österreich
Schweiz

€ 17,70
€ 19,-
sfr 23,-

1

DIGITAL
PRODUCTION

DIGITAL PRODUCTION

MAGAZIN FÜR DIGITALE MEDIENPRODUKTION

JANUAR | FEBRUAR 01:2018



Finishing

IMF, HDR, Retusche
und mehr fürs Finale

Crash! Bam! Pft!

Foleys, Fairlight und Libr-
aries im Audioschwerpunkt

Workshops!

Clarisse, Fusion, Flame,
Lumion, Tracking ...



4 194336 217709





Simulationen mit Höchstleistung berechnen

High Performance Computing (HPC) spielt bislang vor allem im wissenschaftlichen Bereich zur Berechnung, Modellierung und Simulation von komplexen Systemen und zur Verarbeitung von riesigen Messdatenmengen eine große Rolle. Innerhalb des „Media Solution Center“-Projekts beschäftigt sich Dr. Ari Harjunmaa mit der Frage, wie sich Höchstleistungsrechner für Animationen und visuelle Effekte in der Filmindustrie nutzen lassen.

von Mirja Fürst

In der CG-Filmindustrie benötigen zwei Aufgaben extrem viel Rechenleistung: High-Resolution-Simulationen und Bildsynthese (Rendering), wofür zukünftig Hoch- und Höchstleistungsrechner genutzt werden sollen. So würden sich Renderprozesse erheblich beschleunigen, sodass ein datenintensives Rendering nicht mehr nach

mehreren Tagen, sondern innerhalb von Stunden oder sogar Minuten fertig ist.

Das Media Solution Center Baden-Württemberg (MSC-BW) ist Bestandteil einer Initiative des Ministeriums für Wissenschaft, Forschung und Kunst Baden-Württemberg, mit der die Kooperation zwischen der Forschung und der Industrie im Bereich des

HPCs gefördert werden soll. Zukünftig soll das MSC-BW eine Schnittstelle zwischen technisch-wissenschaftlich orientierten Forschungseinrichtungen und kreativschaffenden Medienunternehmen bilden. Mehr Informationen zu dem Projekt gibt es auf: msc-bw.de. Dafür arbeiten Dr. Ari Harjunmaa und sein Team von der Filmakademie



Der Hazel Hen Cray XC40 Flugschiff-Supercomputer im HLRS

Bilder: Filmakademie Baden-Württemberg/HLRS

mit dem Höchstleistungsrechenzentrum Stuttgart (HLRS) und verschiedenen anderen lokalen Medienpartnern wie der Hochschule der Medien und dem Zentrum für Kunst und Medientechnologie in Karlsruhe zusammen.

Dr. Ari Harjunmaa hat Physik und theoretische Physik an der Universität von Helsinki in Finnland studiert, dort hat er 2004 seinen Master und im Jahr 2011 seinen Doktor gemacht. Seine Dissertation befasste sich mit rechen technischer Materialwissenschaft (atomistische Simulationen). Nach der Promotion setzte er von 2012 bis 2014 ein Projekt zu atomistischen Simulationen an der Ruhr-Universität Bochum um.

Mit diesem umfangreichen Background zum Thema physikalische Simulation hat sich Ari im Jahr 2015 an der Filmakademie für seine jetzige Forschungsposition beworben und wurde angenommen.

HPC versus Cloud-Computing

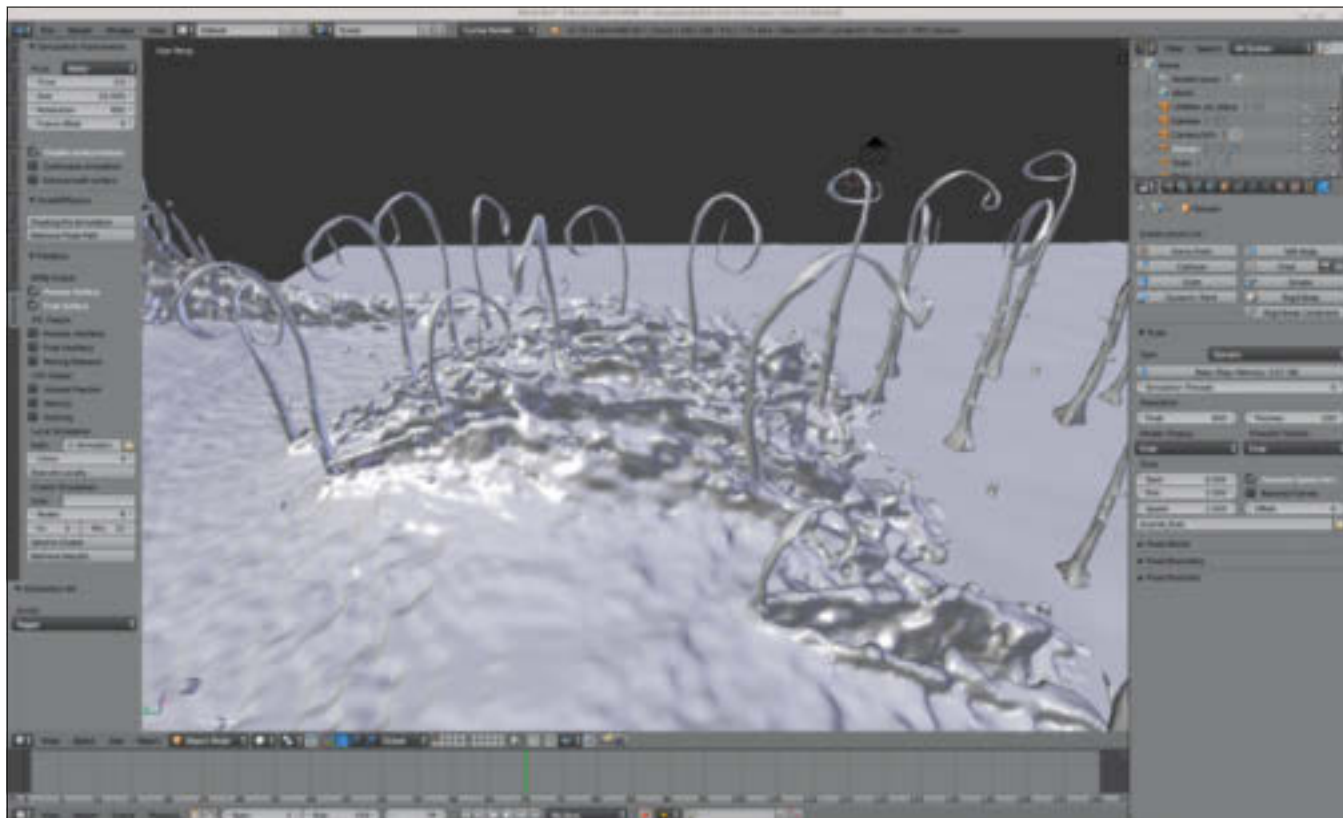
Da HPC seinen Ursprung im wissenschaftlichen Rechnen hat, steht hierfür im Gegensatz zum öffentlichen Cloud-Computing kaum Dokumentation zur Verfügung. Ein weiterer entscheidender Unterschied ist die Homogenität des Systems. Bei den Amazon Web Services können Nutzer den Maschinentypus und die Konfigurationen für jeden Node individuell aufsetzen, jeder Kunde kann die Hardware beeinflussen und besitzt die volle Kontrolle und administrativen Rechte über das Operating System auf den Nodes.

All diese Dinge sind im HLRS nicht der Fall, weshalb Fachleute aus der Medienbranche, die kein HPC-Hintergrundwissen besitzen, nicht wissen, wie bestimmte Aufgaben angegangen werden können.

Braune Superhenne

Der Flugschiff-Supercomputer, der für das MSC-BW-Projekt zum Einsatz kommt, trägt den Namen Hazel Hen. Er ist ein Cray XC40, der auf 7.712 Compute Nodes läuft (entspricht 185.088 Cores) und dessen Spitzenleistung bei 7.4 PetaFLOPS liegt. Auf der letzten Top 500-Liste (www.top500.org/list/2017/06) belegte er im weltweiten Vergleich Platz 17. Mehr Informationen zu dem Supercomputer gibt es auf: bit.ly/Cray_XC40.

Hazel Hen verwendet eine spezielle Cray-Verbindung für die Kommunikation der Nodes. Diese funktioniert schneller als ein konventioneller Cluster, bei dem Desktops parallel geschaltet und über ein Netzwerk miteinander verbunden sind. In der Hazel-Hen-Struktur formen 4 Berechnungsknoten eine „Blade“, die sich einen einzigen



Simulationsszene in dem vom Filmakademie-Forschungsteam entwickelten Prototypen, der auf dem Supercomputer Hazel Hen läuft.

Aries-Network-Chip teilen. 16 dieser Blades sind in einem Chassis montiert und über die sogenannte Backplane all-to-all verbunden. 3 dieser Chassis sind in einem Schrank montiert, 2 dieser Schränke, also 6 Chassis, bilden eine Schrankgruppe. Eine Verbindungen innerhalb dieser Gruppen erfolgt über Kupferkabel, die Schrankgruppen wiederum sind über optische Kabel miteinander verbunden.

Die Kommunikation erfolgt durch eine Cache-kohärente Übertragung von Daten aus dem Hauptspeicher des Source Nodes in den Hauptspeicher des Ziel-Nodes, dabei muss der Aries-Chip die logische Adresse in eine physikalische Speicheradresse übersetzen. Der Speicher wird vom System über Memory Pages verwaltet. MPI (Message Passing Interface) ist ein Standard, der von den meisten Distributed-Memory-Parallelanwendungen genutzt wird.

Es ist eine Bibliothek von C/C++ (oder Fortran) Funktionen, die innerhalb eines Programms aufgerufen werden, damit Aufgaben in Unteraufgaben verteilt werden, sodass sie auf einer Anzahl von einzelnen Kernen laufen. Außerdem realisiert sie die Kommunikation zwischen den Teilaufgaben, um die Kohärenz zu gewährleisten und Ergebnisse von einzelnen Teilaufgaben zu sammeln, sodass ein einzelnes Set von Ergebnissen für die Präsentation gebildet wird.



MPI-Library

Programme, die unter Verwendung von MPI Function Calls geschrieben wurden, müssen mit einem MPI-Compiler kompiliert und mit einem MPI-Launcher ausgeführt werden. Außerdem müssen sie separat installiert werden, bevor die MPI-Programme verwendbar sind – aber die meisten MPI-Implementierungen sind Open Source und lassen sich

Overhead ist aufgrund der Parallelisierung stark reduziert, sodass Simulationen mit niedrigerer Auflösung noch schneller laufen als mit einem herkömmlichen Cluster, ganz zu schweigen von einer Cloud-Computing-Lösung. Dies ermöglicht eine erheblich größere Auflösung von Simulationen, die das Hauptziel unseres Projekts ist und zuvor nicht einmal in einer Compute Cloud möglich gewesen wäre.“

»Für parallele Anwendungen auf Supercomputern liegt in der Filmindustrie ein Mangel vor.«

Dr. Ari Harjunmaa
Physiker, Filmakademie Baden-Württemberg

deshalb frei verwenden. Cloud-Computing ist hauptsächlich für eine perfekte parallele Arbeitsauslastung gedacht, also für mehrere serielle Jobs wie das Rendern einer großen Anzahl von Einzelbildern.

Ari erklärt, warum MPI nur im Umfeld des Hochleistungsrechnens Sinn macht: „Die Verwendung von MPI in einem Cloud-Hosting-Service wie EC2 von Amazon ist nicht unmöglich – aber aufgrund einer fehlenden engen Internode-Kommunikation ist die Obergrenze der Größe eines Systems, mit dem sich effizient simulieren lässt, ziemlich niedrig. Ein Supercomputer weist dagegen eine sehr schnelle lokale Verbindung für die Kommunikation zwischen Nodes auf. Der

Palander v1

Ziel des Projekts Palander v1 war die Erstellung eines grafischen User Interfaces (GUI) für Artists, mit dem sich künstlerischer Content an eine

externe Simulation Engine, die auf dem Hazel Hen Supercomputer läuft, weitergeben lässt. Um dies zu erreichen, untersuchte das Forschungsteam zunächst eine Reihe von Fluid Simulation Engines, die Open Source sind. Man entschied sich für die Nutzung von Palabos, die eine Bibliothek von Lattice Boltzmann Method (LBM) Algorithmen ist. „Von allen Open-Source-Alternativen, die wir zu Beginn des Projekts untersucht haben, stellte sich Palabos als die am besten geeignete heraus, weil sie bereits mit MPI parallelisiert und fähig ist, auf einem anderen Supercomputer mit über 16.000 Cores zu arbeiten. Dies wurde durch zahlreiche Tests, eine sorgfältige Programmierung, eine

intuitive Lastverteilung und Optimierungsalgorithmen erreicht“, so Ari.

Normalerweise müssen die ausführbaren Dateien der Simulation für jeden Zweck separat erstellt werden – entweder von Grund auf oder basierend auf Beispielprogrammen. Für das Projekt benötigte das Team also ein generalisiertes Simulationsprogramm, das abhängig von der Szeneneinstellung für eine Vielzahl von Zwecken einsetzbar ist und mit dem sich eine Simulation unter der Verwendung der externen Parameter ausführen ließ. „Für die Entwicklung des Programms musste ich eine Reihe von meist inkompatiblen Beispielalgorithmen für Basisfunktionalitäten wie Free-Surface-Simulationen und die Auswirkung von Bewegungshindernissen kombinieren, weil es eine oder mehrere dieser Funktionalitäten nutzen würde, abhängig von den übergebenen Parametern. Es fühlte sich an, als würde man ein Puzzle mit Stücken bauen, die nicht zusammenpassen – aber am Ende habe ich es zum Laufen gebracht“, lacht Ari.

Out of the Box funktioniert Palander nur mit Blender und Palabos. Aktuell gibt es keine Pläne, kommerzielle DCC-Software in den Prototypen zu integrieren – zwar bestand die Möglichkeit immer, aber es gab bislang keinen Bedarf dafür. Ari dazu: „Das Konzept 'DCC-Add-on plus externe Engine' würde natürlich für jede DCC-Software-Engine-Kombination funktionieren und es kann auch auf der Hazel Hen laufen, solange die betreffende Engine ordentlich parallelisiert ist. Innerhalb von Palander sind die Add-on- und Simulations-Engine separate Programmdateien, weshalb sie individuell ersetzt werden könnten – beispielsweise mit

einem Add-on für Maya, das dieselben XML- und STL-Parameterdateien wie die aktuelle Version ausgeben kann.“

Palander wurde auf der FMX 2017 veröffentlicht und während einer Präsentation vorgestellt. Als Prototypen-Entwickler hat das Forschungsteam den Vorsprung für Filmakademie-Projekte ausgenutzt: Palander wurde unter anderem für die studentische Produktion des offiziellen Trailers der FMX 2018 „Biber im Bart“ genutzt. Für das Projekt ist eine Trailer-Folge geplant, die mit Palander simuliertes Wasser enthält. Auch der Projektpartner ZKM setzt Palander ein, um in Zusammenarbeit mit dem österreichischen Künstler Helmut Schober einen Wasserwirbel für ein Kunstprojekt zu simulieren. Des Weiteren hat das Team in Zusammenarbeit mit dem Simulationsteam von Mackevision die Ergebnisse einer Palander-Simulationsszene mit denen einer mit Phoenix auf Cinema 4D simulierten verglichen.

Blender-Interface

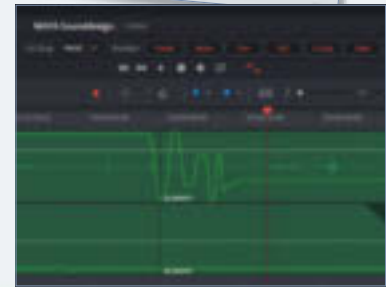
Der nächste Schritt beinhaltet die Auswahl eines Open-Source-GUIs. Das Team entschied sich aufgrund seiner umfangreichen Funktionalitäten für Blender. „Blender besitzt dasselbe Niveau wie die meisten kommerziellen DCC-Tools, die es einfacher machen würden, unseren Prototypen in diese anderen Tools zu konvertieren, wenn es notwendig wäre“, meint Ari. Nachdem das Team Blender gelernt hatte, schrieben sie ein Add-on, mit dem die relevanten Simulationsparameter an einen externen Solver übergeben werden konnten. Das Blender-Add-on-Menü beinhaltet Settings für einige der Simulationsparameter, die nicht in Blenders eigenem Parameterset enthalten sind, sowie verschiedenen Funktionen, durch die Simulationen sowohl auf dem lokalen Desktop als auch auf dem Supercomputer Hazel Hen laufen können.

Blenders native Fluid Simulation Engine wird dabei mit einem Python-Add-on umgangen, welches die relevanten Simulationsparameter aus den Blender-eigenen Szenenvariablen sowie dem Add-on-Menü übernimmt und in eine XML-Datei einfügt. Zusätzlich exportiert es alle STL-Dateien aller aktiven Fluid- oder Rigid-Objekte in der Szene. Anschließend werden diese Dateien an das lokale Simulationsverzeichnis oder Hazel Hen übergeben und ein Skript ausgeführt, um die Flüssigkeitssimulation mit der Palabos-Datei zu starten, die für diesen Zweck erstellt wurde. Wird sie lokal ausgeführt, gibt sie die Simulationsergebnisse direkt in das Simulations-Cache-Verzeichnis von Blender aus, sodass die Ergebnisse während der Produktion angezeigt werden können.



Der geschenkte Gaul

Ein High-End-DAW für Lau? Wir schauen uns den neuesten Reiter in Resolve an – und schauen, ob es noch ein echtes Fairlight ist. Und natürlich, was jetzt schon möglich ist. Das und mehr lesen Sie im kostenlosen Artikel, zum Download unter: www.digitalproduction.com



CREATIVE MEDIA EDUCATION

Diploma-, Bachelor-
und Masterabschlüsse
in der Medienbranche

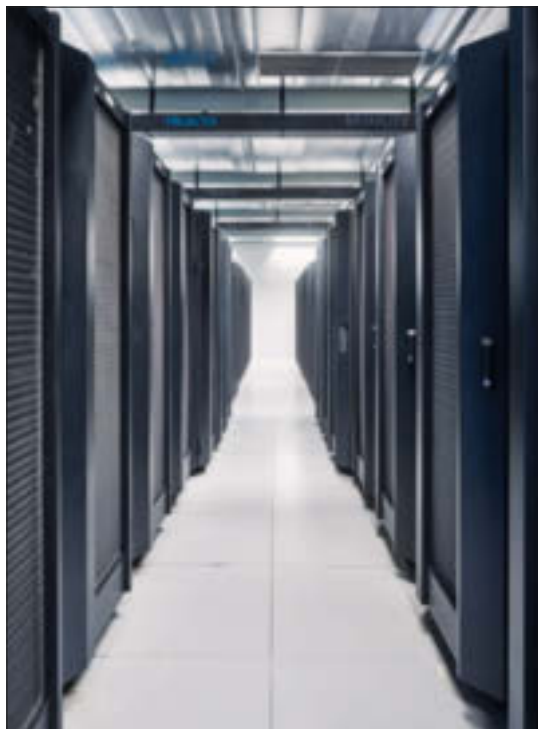
www.sae.edu

Dies ahmt die Funktionsweise von Blenders nativer Simulations-Engine nach, während es diese effektiv umgeht. Durch intensives Testen verfeinerte das Team den Add-on- und Engine-Prototypen so lange, bis er auf zufriedenstellende Weise arbeitete. Mithilfe des Prototypen produzierten die Forscher Simulations-Content, der dann mit dem aus professionellen Produktionen verglichen wurde.

MSC-BW für die CG-Branche

Zukünftige Pläne für das MSC-BW-Projekt beinhalten die Gründung eines Vereins, der von lokalen VFX- und Animationsfirmen mitfinanziert werden könnte. Das MSC-BW wird dann Beratungsdienste anbieten, durch die sich auf die individuellen Kundenbedürfnisse zugeschnittene Lösungen finden lassen. Diese können Palander und/oder eine maßgeschneiderte Variation davon sowie den Zugriff auf den Supercomputer beinhalten.

Den Palander-Code in seinem aktuellen Zustand hat das Team als Open-Source-Programm auf GitHub veröffentlicht (github.com/FilmakademieRnd/Palander). Er kann frei heruntergeladen und auf einem Computer implementiert werden. Um eine Simulation auf Hazel Hen laufen zu lassen, benötigt man jedoch ein HLRS-Konto. Durch geringfügige Code-Modifikationen lassen sich die Simulationen so anpassen, dass sie auch auf anderen Supercomputern oder Computerclustern laufen können.



In seiner aktuellen Version ist Palander in der Lage, eine beliebige Anzahl von flüssigen Volumen und Hindernissen jeder Form innerhalb einer einzigen quaderförmigen Domäne zu simulieren. Zu- und Abflüsse sowie bewegliche Hindernisse, die auf das Wasser treffen, sind implementiert. Zusätzlich zum Blender-lesbaren Output kann Palander STL- und/oder VTK-Dateien der Simulationen erzeugen. Letztere können Geschwindigkeitsinformationen enthalten, die beispielsweise für Vertex Coloring, das sich zum Erzeugen von Schaum eignet, nützlich sein können.

Das HLRS arbeitet bereits mit zahlreichen Kooperationspartnern aus verschiedenen Branchen zusammen, wie zum Beispiel der Automobil-, der Medizin-, Energie- und der

Stadtplanungsindustrie; die mit der Filmindustrie ist dagegen neu. Vor ein paar Jahren nutzte M.A.R.K.13 den Supercomputer des HLRS für das Rendering von einem Großteil der Frames für den Animationsfilm "Biene Maja" (bit.ly/BieneMaja_HLRS). Dabei kam jedoch eine mehrfache serielle Anwendung ohne die Parallelisierungsfunktionen eines Supercomputers zum Einsatz.

Für parallele Anwendungen, bei denen Supercomputer zum Simulieren oder Rendern verwendet werden, liegt ein offensichtlicher Mangel an kommerziellen Lösungen in der Filmindustrie vor. Selbst wenn es solche Lösungen gäbe, würden sie in der Regel eine separate Lizenz zum Ausführen auf jedem Supercomputer-Node erfordern, was den Prozess unerschwinglich teuer macht. „Für dieses Problem existieren zwei Ansätze: Entweder wir lassen mehrere Tage lang Simulationen auf Desktops mit kommerzieller Software laufen oder verwenden Open-Source-Software auf einem Cluster oder Supercomputer, um die Berechnungszeit von Tagen auf Stunden oder sogar Minuten zu reduzieren. Unser Projekt zeigt, dass Letzteres eine Möglichkeit ist – es liegt in der Verantwortung der Industrie, es zu nutzen. Die Rolle der HPC-Technologie für Animationen und Effekte wird in naher Zukunft immer wichtiger werden, wenn Studios das Potenzial der hohen Rechenleistung für sich entdecken. Gerüchten zufolge hat Weta Digital bereits den Kauf eines Cray-Supercomputers in Erwägung gezogen und ihre Software dahingehend weiterentwickelt. Viele große Studios, die sich die Hardware-Investition leisten können, werden zweifellos folgen. Kleineren Konkurrenten können dagegen erschwingliche und attraktive Angebote des MSC-BW nutzen“, resümiert Ari. > mf



Außenansicht des HLRS-Gebäude in Stuttgart

Links

- MSC-BW-Projekt
▷ m-sc-bw.de
- Supercomputer-Ranking
▷ www.top500.org/list/2017/06
- Cray XC40 Hardware & Architektur
▷ bit.ly/Cray_XC40
- Palander auf Github
▷ bit.ly/FilmAkaPalander
- Palander Fluid Simulation
▷ bit.ly/palanderFILMAKA
- Filmakademie Baden-Württemberg
▷ www.filmakademie.de