

CPU-Leistungsoptimierung durch Prozess-Bindung

Im Technical Computing ist die CPU-Leistung einer der begrenzenden Faktoren. Dieser Artikel handelt von der gezielten Bindung von Berechnungsprozessen an Prozessoren. Diese Bindung ermöglicht es, die Performance von Linux Compute-Servern mit Non-Uniform Memory Access (NUMA) Speicher-Architektur zu steigern.

Mit einer Prozess-Bindung sind bei CFD- und Crash-Applikationen Performance-Verbesserungen von ca. 15% oder mehr möglich. Insbesondere trifft dies für AMD Opteron Server zu. Der Text vermittelt auch die Anforderungen an eine Prozess-Bindung beim produktiven Betrieb von Compute-Clustern mit einem Batch-Queueing-System.

Bei NUMA Speicher-Architekturen ist jedem Prozessor lokaler Speicher mit hoher Bandbreite und geringer Latenz zugeordnet. Speicher anderer Prozessoren werden indirekt mit einem oder mehreren Sprüngen über den Hypertransport-Bus angesprochen, was die Latenz dieser Speicherzugriffe erhöht (Abb. 1 a+b). Bei Multicore-

Prozessoren teilen sich die Cores den lokalen Speicher, so dass indirekter Speicherzugriff erst bei mehreren Prozessoren auftritt.

Ein speicherintensiver Solver-Prozess einer CAE-Applikation ist also dann am schnellsten, wenn er möglichst viel lokalen Speicher seines Prozessors benutzt. Wird ein Prozess nach dem ersten Speicherzugriff vom Kernel auf einen anderen Prozessor migriert, bleibt der Speicher zurück und die Zugriffe des Prozesses sind nun indirekt und langsam. Für beste Performance ist es also meist sinnvoll, den Wechsel von Prozessen zwischen Prozessoren zu unterbinden.

Linux unterstützt die NUMA Speicherarchitektur ab Kernel 2.5.8-pre3 mit System-Aufrufen zum Manipulieren der CPU-Affinität eines Prozesses. Für ältere 2.4er Kernel gibt es hierzu Patches. Unter anderem unterstützen die Distributionen Red Hat 9 und SuSE 9.0 bereits die CPU-Affinität.

Prozess-Bindung hält auch Einzug bei einigen Varianten des Message Passing Interfaces (MPI). So kann bei OpenMPI mit der Option `mpirun --mca mpi_paffinity_alone 1` Prozess-Bindung

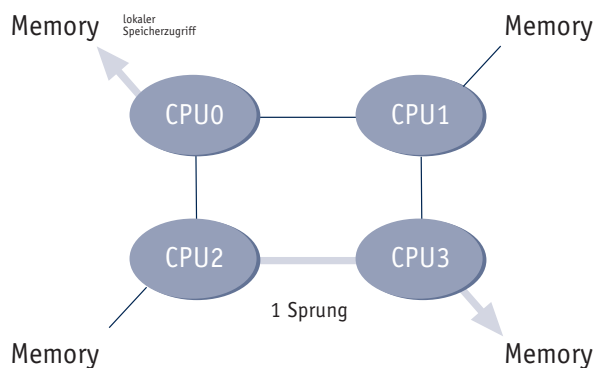


Abb. 1a: Direkte Speicherzugriffe über den Hypertransport-Bus bei vier NUMA Prozessoren

News

Am 1. August 2006 hat GNS mbH eine Geschäftsstelle in Melbourne, Australien eröffnet.

Beiträge

CPU-Leistungsoptimierung durch Prozess-Bindung

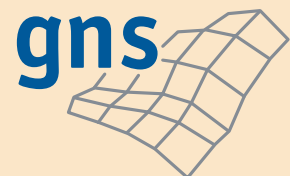
Player und Powerpoint Plugin für Animator3

FETI-INDEED - Teil 1

Verteilte Umformsimulation auf Compute Clustern

CA-Systemservice als externe Dienstleistung

Videofunktionalität in Animator3



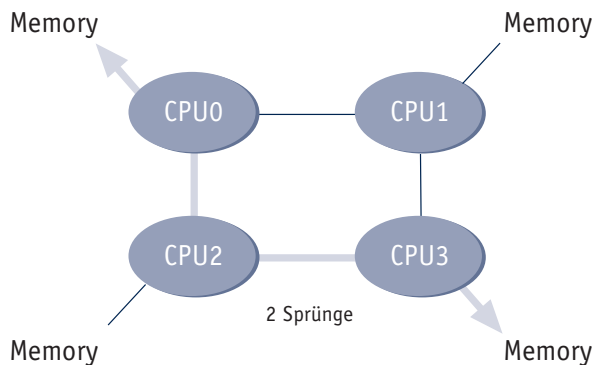


Abb. 1b: Indirekte Speicherzugriffe über den Hypertransport-Bus bei vier NUMA Prozessoren

aktiviert werden (immer Unterstützung durch den Kernel vorausgesetzt). Hier wird allerdings angenommen, dass der Berechnungs-Job auf den jeweiligen Maschinen exklusiv ist. So kann es bei mehreren Jobs ungünstigerweise zu Mehrfachbelegungen einiger Prozessoren kommen, während andere untätig bleiben. *HP-MPI* hat hierzu die Option `mpirun -cpu_bind=<params>`. Seit der Version *HP MPI 2.2* können damit diverse Strategien zur Prozess-Bindung angewendet werden.

Bei den meisten Software-Produkten hat man allerdings keine freie Auswahl des Message Passing Interface. Eine Lösung zur Prozess-Bindung sollte also mit allen MPI-Varianten zusammenarbeiten können. Außerdem sollen natürlich auch Single-CPU Berechnungs-Jobs an einen Prozessor gebunden werden können.

Um die unveränderten Solver-Prozesse der Applikationen zu platzieren, bieten sich die verschiedenen Linux-Kommandos zur Prozess-Bindung an. Je nach Distribution und Kernel-Version heißen sie `numactl`, `taskset`, `dplace` oder `oncpu`. Eine Prozess-Bindung kann einfach mit Shell-Skripten unter Verwendung dieser Kommandos implementiert werden.

Bei einem produktiven Einsatz in einem Linux-Cluster mit unterschiedlichen Applikationen und einem Batch-Queueing-System wie z.B. *Platform LSF* sind neben den oben erwähnten Aspekten weitere Anforderungen zu beachten. Einige Applikationen, wie z.B. *Powerflow*, haben neben den speicherintensiven Solver-Prozessen auch einen Kontrollprozess, der wenig CPU-Last erzeugt und gerade nicht gebunden werden soll. Wünschenswert ist also die Möglichkeit, nur bestimmte Prozesse zu binden.

Ein Batch-Queueing-System soll einen Compute-Cluster ideal auslasten. Dabei können durchaus mehrere Jobs auf einem Cluster-Host nebeneinander laufen. Hier ist es wichtig, dass keine Mehrfach-Bindung an einen Prozessor erfolgt. Gerade auch bei gleichzeitiger interaktiver Nutzung der Cluster-Hosts und Job-Verdrängungsmechanismen (Job Preemption) sollen nie mehr Prozesse gebunden sein als Prozessoren im jeweiligen Host verfügbar sind. Es ist also unerlässlich, dass ein Tool zur Prozess-Bindung über die gebundenen Prozesse Buch führt und sich auch bei quasi zeitgleichen Aufrufen nicht durcheinander bringen lässt.

Fazit: Mit Mitteln aktueller Linux Betriebssysteme und etwas Scripting kann eine Prozess-Bindung für Compute-Server mit NUMA Speicherarchi-

Bildergalerie

VDI-Tagung, September 2006, im Maritim Hotel in Würzburg
Berechnung & Simulation im Fahrzeugbau



Claudius Schöne (GNS mbH), Volker Elß (GNS mbH), Philipp Nolden (GNS Systems GmbH), Koutaiba Kassem-Manthey (GNS mbH) (von links nach rechts)

LS-DYNA Forum, Oktober 2006, im Maritim Hotel in Ulm



Jan Martini, GNS Systems GmbH (links), Joachim Wagner, GNS Systems GmbH und GNS mbH (rechts)

tektur umgesetzt werden. Damit ist eine Performance-Verbesserung speicherintensiver Berechnungs-Jobs von ca. 15% möglich.

Marcus Renner, GNS Systems GmbH
marcus.renner@gns-systems.de

Player und Powerpoint Plugin für Animator3

Um die Produktivität unserer *Animator3*-Kunden zu erhöhen und den Datenaustausch zwischen Kunden und Zulieferern zu erleichtern, hat *GNS* für *Animator3* gerade ein kostenloses Werkzeug entwickelt. Dieses neue Programmpaket stellt *Animator3*-Datenbanken dar und eignet sich hervorragend für Vertrieb und Marketing. Da es keinen Kontakt zum Lizenzserver benötigt, kann es sehr gut im mobilen Einsatz auf dem Laptop genutzt werden. Beim Einsatz in verteilten Projektteams ermöglicht es den einfachen und schnellen Datenaustausch mit Partnern und entfernten Kollegen, die über keine *Animator3*-Lizenz verfügen, sich aber an einem 3D-Modell schnell und einfach einen Überblick über ein gemeinsam bearbeitetes Bauteil verschaffen wollen. Durch die Möglichkeit, die stark komprimierte *Animator3*-Datenbasis einzulesen, müssen nur geringe Datenmengen transportiert werden.

Die Vorbereitung der *Animator3*-Datenbasis geschieht in einer normalen *Animator3*-Sitzung, d.h. die Eigenschaften des Animationsmodells werden dort für die Speicherung in einer *Animator3*-Datenbasis definiert. So kann der Benutzer z.B. die Farben der unterschiedlichen Properties verändern oder Gruppen von Bauteilen und Elementen definieren, die dann später

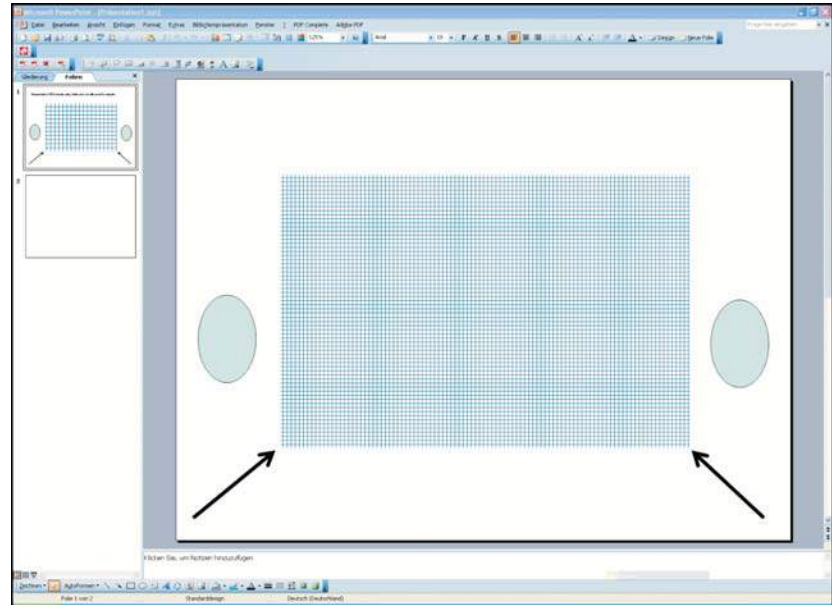


Abb. 1: Darstellung von FEA-Ergebnissen mit *Animator3* und *Microsoft Powerpoint*

mit dem Player angesprochen werden. Es ist möglich, das Modell wie in *Animator3* zu drehen, für Detailbetrachtungen in das Modell hineinzuzoomen und einzelne Gruppen und Elemente ein- und auszublenzen. Änderungen an Farben und Darstellungsmodi sowie die Definition von Gruppen können mit dem Player jedoch nicht mehr vorgenommen werden.

Nach der Installation des Active-X-Plugins auf einem Windows-Rechner taucht das Plugin als 'A3_ax Control' in der Liste der unterstützten Objekte zum Einfügen in *Powerpoint* auf. In der Übersicht erscheint die für das Plugin reservierte Fläche als

Gitter; andere Elemente können so um das Plugin herum arrangiert werden (Abb. 1). Mit der rechten Maustaste wird das Eigenschaften-Menü aufgerufen. In diesem Menü kann die Hintergrundfarbe des Plugins gesetzt und die Database-Datei angegeben werden, die beim Start der Präsentation eingelesen wird. Schaltet man dann auf den Präsentationsmodus um, stellt das A3_ax-Plugin ein 3D-Modell für die Präsentation zur Verfügung (Abb. 2).

Der *Animator3-Player* wird als ActiveX-Plugin für *Microsoft-Powerpoint* sowie als stand-alone Executable für *Windows*, *LINUX* und sämtliche von *Animator3* unterstützte *UNIX*-Varianten zur Verfügung gestellt. Der Player steht auf der *GNS*-Webseite (www.gns-mbh.com) zum Download bereit.

Christoph Kaulich, *GNS mbH*
animator@gns-mbh.com

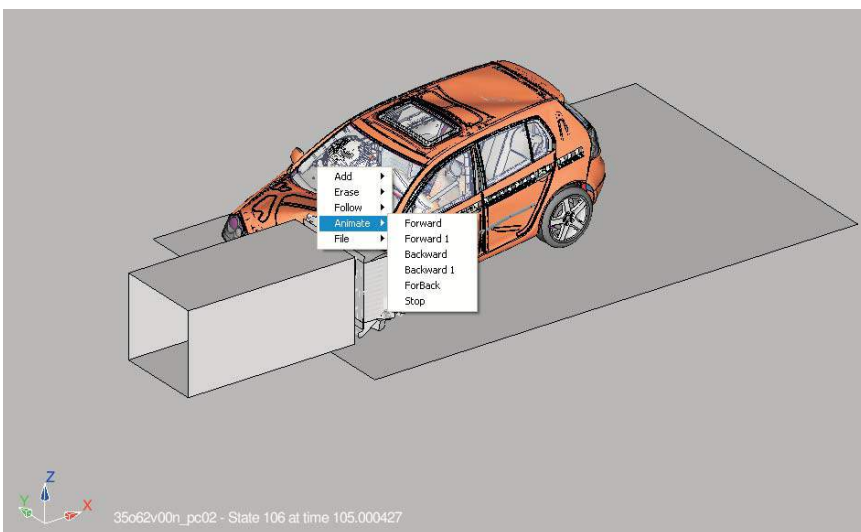


Abb. 2: *Animator3 Powerpoint-Plugin* mit eingelesenem Modell

FETI-INDEED - Teil 1

Verteilte Umformsimulation
auf Compute Clustern

Die Finite-Element-Software *INDEED*® kommt zum Einsatz, wenn hochgenaue Ergebnisse bei der numerischen Simulation von Blechumformvorgängen gefordert werden. Ohne Frage hat Ergebnisqualität ihren Preis, einerseits in Form von längeren Rechenzeiten, andererseits aber auch in Form eines erhöhten Speicherplatzbedarfes. Und dennoch: es besteht kein Zweifel

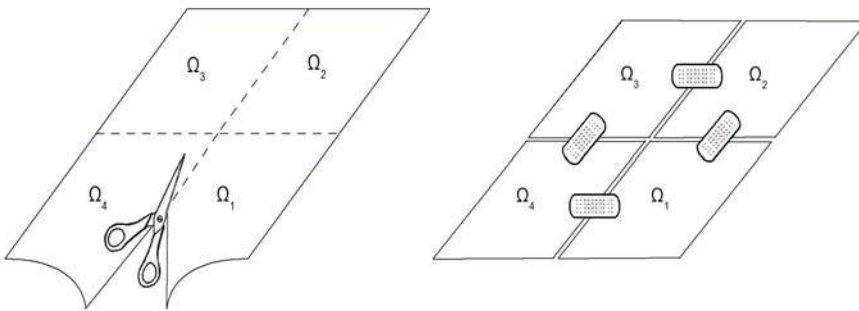


Abb. 1: Prinzipielle Darstellung der FETI-Methode

daran, dass die Ansprüche an die Genauigkeit von Simulationsergebnissen ständig weiter steigen werden. Eine Folge davon wird die Verwendung immer komplexerer Materialmodelle und feinerer Finite-Element-Netze mit höherwertigen Elementtypen sein.

Steigende Genauigkeit wird aber auch eine möglichst lückenlose Abbildung der gesamten Prozesskette bedingen. Damit wird in Zukunft der Rechenaufwand und Speicherbedarf noch deutlich über dem liegen, was heute üblich ist. Es ist daher unumgänglich, eine Vielzahl von Prozessoren effektiv einzusetzen, um Simulationen trotz des steigenden Rechenaufwandes auch weiterhin kosteneffizient durchführen zu können.

INDEED wird heute zumeist noch in der SMP-Version (Shared Memory Processing) auf Mehrprozessormaschinen verwendet. Ein großer Nachteil des Shared Memory Processings ist, dass die Skalierbarkeit hierbei schnell an ihre Grenzen stößt. So ergibt sich bei Ausführung der SMP-Ver-

sion von *INDEED* eine Skalierung bis maximal acht Prozessoren, danach ist kein nennenswerter Rechenzeitgewinn mehr festzustellen. Der Einsatz weiterer Prozessoren ist also wirtschaftlich nicht mehr sinnvoll. Zudem muss beim Shared Memory Processing der Hauptspeicher so dimensioniert sein, dass auch für sehr große Modelle, wie sie z.B. bei der Tiefziehsimulation eines kompletten Fahrzeugseitenteils erforderlich sind, alle Berechnungsdaten komplett hineinpassen. Dies ist notwendig, um ein zeitintensives „Swappen“ zwischen Hauptspeicher und Festplatte zu vermeiden. Deshalb steht

für *INDEED* neben der SMP-Version eine spezielle Version für den Einsatz auf Compute-Clustern zur Verfügung: *FETI-INDEED*.

Die Idee hinter *FETI-INDEED* besteht darin, das Gesamtproblem in mehrere, möglichst unabhängige Teilprobleme aufzuspalten. Dies geschieht durch die Zerlegung des entsprechenden Finite-Element-Modells in mehrere Untergebiete, die sogenannten Subdomains. *FETI-INDEED* basiert also auf einem speziellen Gebietszerlegungsverfahren. Dieser Domain-Decompositioning-Ansatz erlaubt es, alle Phasen der Simulation (Aufbau des Gleichungssystems, Lösen und Linearisierung, Nachlaufrechnung) auf den verwendeten Prozessoren bzw. Rechnern eines Clusters nahezu unabhängig voneinander durchzuführen. Bei der Wahl des Gebietszerlegungsverfahrens ist jedoch darauf zu achten, dass der Informationsaustausch nur zwischen unmittelbar benachbarten Subdomains erfolgt und eine schnelle Konvergenz des Verfahrens auch bei hochgradig nichtlinearen Problemen gegeben ist. Nur so lässt

sich eine gute Skalierbarkeit auch bei einer sehr großen Anzahl von Prozessoren erreichen. Als Domain-Decompositioning-Verfahren für *INDEED* wurde die sogenannte FETI-Methode gewählt, die sich nach entsprechenden Anpassungen auf die speziellen Erfordernisse von *INDEED* als besonders gut geeignetes Verfahren erwiesen hat.

Die FETI-Methode

Die „Finite Element Tearing and Interconnecting“-Methode (FETI) von Charbel Farhat und François-Xavier Roux stellt eine Möglichkeit dar, selbstadjungierte Differentialgleichungen mittels der Finite-Elemente-Methode parallel zu lösen. Das zu berechnende Problem wird in mehrere kleinere Probleme aufgespalten, die jeweils von einer eigenen „Simulationsinstanz“ bearbeitet werden können (Abb. 1).

Unter Verwendung der *METIS*-Bibliothek partitioniert *FETI-INDEED* selbstständig die Platine in Teilgebiete (Subdomains), in denen jedoch die Werkzeugnetze sowie die Polygonzüge zur Beschreibung von Ziehsicken, Absperrleisten oder Beschnittlinien komplett vorliegen.

Die Kommunikation, d.h. der Austausch der Informationen, zwischen den Teilgebieten erfolgt ausschließlich über deren Randknoten. Daher muss bei der Partitionierung darauf geachtet werden, die Platine so aufzuteilen, dass die Anzahl der inneren Randknoten minimal wird. Eine weitere Anforderung an die Partitionierung besteht darin, dass die Knoten- bzw. Elementanzahl und somit der Rechenaufwand für alle Subdomains in etwa gleich ist.

Der Partitionierungsprozess funktioniert unabhängig von der räumlichen Anordnung der Knoten, da das Platinennetz in einen



Graphen umgewandelt und dieser dann unter den genannten Vorgaben aufgeteilt wird. Für die Teilung der Platine kann der Graph Informationen bezüglich der Nachbarschaften von Knoten oder Elementen enthalten.

Im Gegensatz zu anderen Partitionierungsalgorithmen verwendet *METIS* die so genannte „Multilevel-Partitionierung“. Hierbei wird versucht, den vorgegebenen Graphen zunächst zu vergrößern, indem Knoten und Kanten des Graphen sukzessive zu einem übergeordneten kleineren Graphen zusammengefasst werden. Der initiale Graph wird dann entsprechend der gewünschten Partitionsanzahl des Benutzers aufgeteilt und auf den Originalgraphen projiziert. Hierdurch wird eine qualitativ hochwertige Partitionierung erreicht, die zudem schnell durchgeführt werden kann. Abbildung 2 zeigt exemplarisch ein in acht Subdomains partitioniertes Praxisbauteil.

Mit der optimalen Aufteilung der Platine werden dann die Eingabedaten für die jeweiligen Subdomains zusammengestellt und die Teilprobleme auf den einzelnen Prozessoren gelöst. Durch die Partitionierung kann es allerdings vorkommen, dass ein Teilgebiet zu einer sogenannten „floating subdomain“ wird. Ein solches Teilgebiet könnte aufgrund fehlender Kontakt- oder Benutzerrandbedingungen Starrkörperbewegungen ausführen. Dieses hätte zur Folge, dass die Systemmatrix in einem Teilgebiet nicht invertierbar und das Gleichungssystem somit nicht lösbar wäre. Dieses Problem wird in der FETI-Methode jedoch automatisch erkannt und mit Hilfe einer so genannten Pseudoinversen gelöst.

Während der Berechnungsphase beeinflussen sich die benachbarten Teilgebiete nur über ihre gemeinsamen Randknoten. Aus diesen ergeben sich zusätzliche Randbedingungen, die dafür sorgen, dass die Lösungen zwischen den Teilgebieten klaffungsfrei ineinander übergehen. Das Erstellen dieser Randbedingungen erfolgt durch ein so genanntes duales Problem, dessen Lösung die Randbedingungen für den nächsten Iterationsschritt festlegt. Da das duale Problem eine Dimension entsprechend der Anzahl der Randknoten besitzt, ist es relativ klein und beansprucht

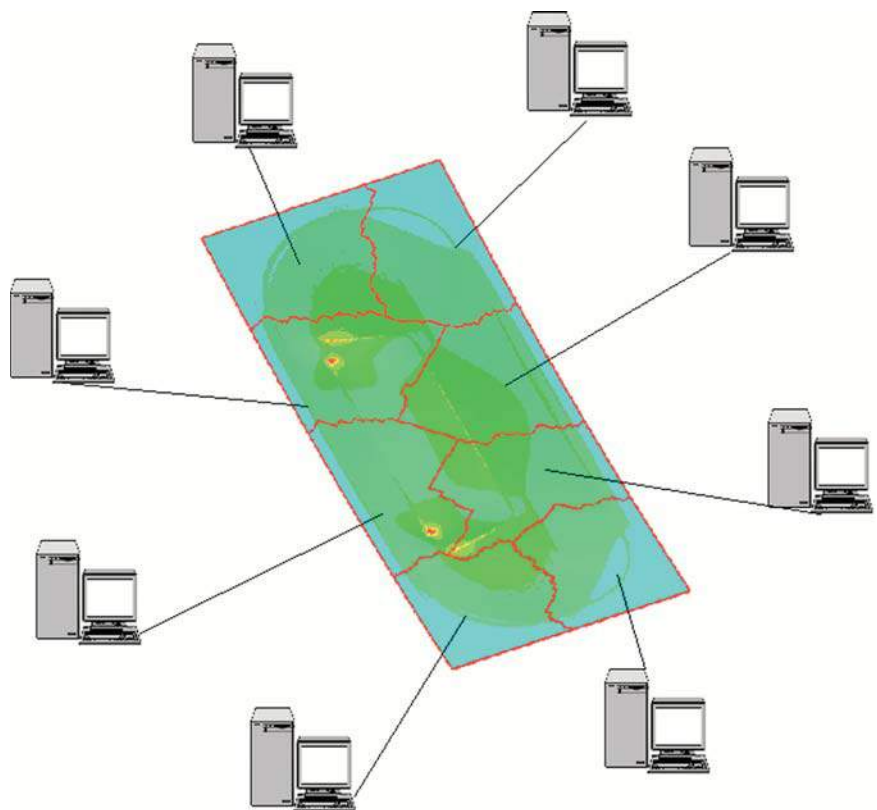


Abb. 2: Beispiel eines in acht Teilgebiete partitionierten Praxisbauteils

sonit nur wenig Rechenzeit. Für den Anwender ändert sich kaum etwas im Vergleich zur sequentiellen oder zur SMP-Version von *INDEED*. Neben der automatischen Partitionierung werden nach der Berechnung die einzelnen Teilgebiete programmintern wieder zu einer Gesamtlösung zusammengesetzt, die er wie gewohnt bewerten kann.

Vorteile der FETI-Methode

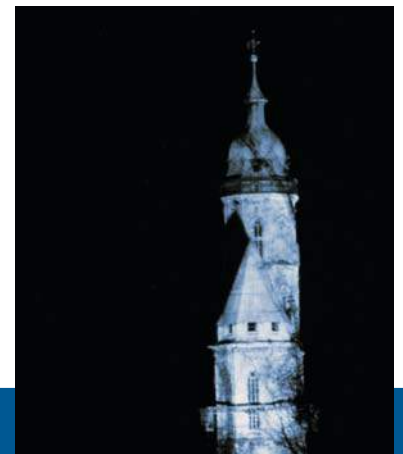
Im Gegensatz zu SMP-Parallelisierungen oder Programmen mit parallelem Gleichungslöser lassen sich mit dem FETI-Ansatz sämtliche Berechnungsschritte parallelisieren. Da zu keiner Zeit das Gesamtproblem betrachtet werden muss, genügt jeweils ein relativ kleiner Arbeitsspeicher auf allen verwendeten Maschinen.

Die Gleichungssysteme der Teilgebiete können nahezu unabhängig voneinander gelöst werden. Dadurch erfordern bei Verwendung der FETI-Methode weder der Aufbau noch die Lösung oder die Nachbearbeitung des Gleichungssystems eine Kommunikation. Lediglich bei der Be-

stimmung des dualen Randproblems werden Daten aus allen Teilgebieten benötigt. Dabei werden jedoch nur Daten zwischen unmittelbar benachbarten Gebieten ausgetauscht. Dieser Kommunikationsaufwand ist verhältnismäßig gering, so dass sich die FETI-Methode in idealer Weise zur Implementierung auf Clustern eignet.

Eine genauere Beschreibung der FETI-Methode und ihrer mathematischen Hintergründe folgt in der nächsten Ausgabe von *GeNius*.

Rene Menzel, GNS mbH
Martin Washausen, GNS Systems GmbH
indeed@gns-mbh.com



CA-Systemservice als externe Dienstleistung

Aus verschiedenen fachlichen, finanziellen und organisatorischen Gründen entscheiden sich große und mittelständische Industrieunternehmen für die Vergabe von IT-Dienstleistungen an externe Partner. Beim CA-Systemservice z.B. handelt es sich um Basisaufgaben innerhalb der Anwendungsbereiche CAD und CAE. Dieser Artikel beschreibt und erläutert fachliche und organisatorische Aspekte von Dienstleistungen in diesem Umfeld, die von einem externen Unternehmen erbracht werden.

Themen

Der externe Dienstleister bearbeitet folgende Themenstellungen in einem mit dem Auftraggeber abgestimmten Umfang:

- Systemmanagement
 - Beschaffung von Workstation- und Serversystemen
 - Betriebssysteminstallation und -konfiguration
 - Netzwerkintegration
 - Datensicherung
 - Systemüberwachung und Performance Monitoring
- Application Management
 - Installation und Konfiguration von CAD- und CAE-Applikationen
 - Betrieb von Lizenz-Servern
- Systemintegration
 - Integration von Anwendungs- und Systemkomponenten
 - Erstellung von Utilities und Zusatzsoftware als Erweiterung von CAD- und CAE-Systemen
- First Level Support
 - Unterstützung bei der Nutzung und Bedienung von Hardware- und Softwarekomponenten.
 - Unterstützung der Anwenderschaft und der Administratoren bei systemorientierten Problemstellungen

Überwachung

Der Dienstleister überwacht kritische System- und Anwendungskomponenten, um deren Funktions- und Leistungsfähig-

keit sicherzustellen. Somit erkennt und verhindert er Fehlfunktionen frühzeitig. Folgende Komponenten sind z.B. für eine Überwachung relevant:

- Systemkomponenten
 - Prozessoren
 - Festplatten
 - Eventlogs
 - Memory
 - Swap Space
 - Netzwerkverbindungen
 - NFS
- Anwendungskomponenten
 - Web Server
 - Datenbanksysteme
 - Lizenzserver

Zwei vielfach in diesem Umfeld eingesetzte Softwarepakete sind *Ganglia* und *Nagios*. *Ganglia* ist besonders für die Überwachung von Hardwaresystemen geeignet (Abb. 1). *Nagios* dient der Zustandsüberwachung von diversen System- und Anwendungsdiensten. *Nagios* ermöglicht es, die Verfügbarkeit von Plattenplatz auf einem File Server oder die generelle Funktionsfähigkeit eines Web Servers zu überwachen. Zusätzlich kann *Nagios* mit definierten Methoden auf bestimmte festgelegte Ereignisse und Situationen reagieren. Hiermit ist es beispielsweise möglich, Nachrichten zu versenden oder diverse Programme anzustoßen.

Die im Rahmen des laufenden Betriebs gesammelten Informationen können als Protokoll verwendet werden, um Verhaltensmuster zu analysieren und relevante Trends zu ermitteln. Die gewonnenen Erkenntnisse fließen dann z.B. in Kapazitäts- und Funktionsplanungen ein.

Help Desk Tools

Bekannte und von den Anwendern gemeldete Probleme sollten in einem Trouble Ticket System erfasst werden, so dass eine Auswertbarkeit und Nachvollziehbarkeit der Störungen gewährleistet ist.

Wichtig bei der Konfiguration eines Ticket Systems ist die Festlegung einer Hierarchie von Themen, nach denen eingehende Meldungen kategorisiert werden. Die Kategorisierung erlaubt eine strukturierte Auswertung, z.B. um festzustellen, welche

Probleme typischerweise bei CAD-Anwendern auftreten. Eine Kategorisierung dient außerdem dazu, Informationen über bereits vorhandene Einträge wieder aufzufinden.

Um den Aufwand insbesondere bei der Problemaufnahme zu reduzieren, sollte der Dienstleister das verwendete Trouble Ticket System so konfigurieren, dass Standardvorgänge schnell bearbeitet werden können. Hierzu muss entschieden werden ob die unterstützten Kunden ggf. direkt Problemstellungen eingeben dürfen. (Abb. 2 zeigt typische Einträge in einem Trouble Ticket System.)

Einfache statistische Auswertungen können über die erfassten Probleme und deren Bearbeitung erstellt werden. Das könnten unter anderem folgende sein:

- Anzahl erfasster Probleme
- Problemschwerpunkte
- Prozentualer Anteil der gelösten Probleme
- Problembearbeitungszeiten
- Organisationseinheiten, von denen Probleme gemeldet werden

Weitergehende Auswertungen sind problematisch, da z.B. Bearbeitungsdauer oder

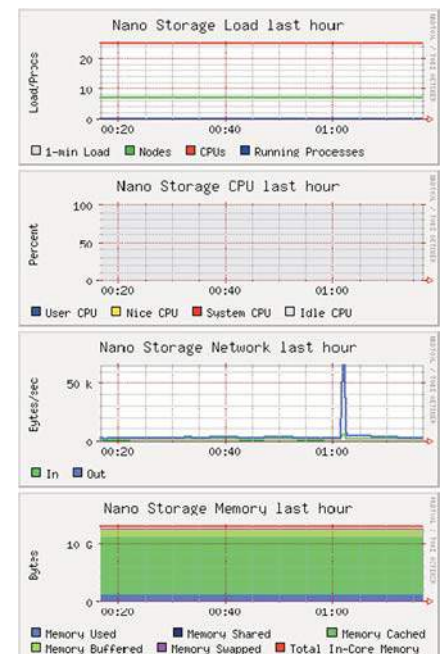


Abb. 1: Zustandsinformationen einer Systemumgebung unter Verwendung von Ganglia



Abb. 2: Typische Einträge in einem Trouble Ticket System

die Anzahl ungelöster Probleme nur schwer bewertet werden können. Obwohl ein Trouble Ticket System vielfältige Informationen enthält, reicht es nicht als ausschließliche Dokumentationsquelle für Administratoren.

Remote Management

Ein überwiegender Teil der im CA-Systemservice anfallenden Aufgaben kann aus der Ferne, vom Standort des externen Dienstleisters aus, bearbeitet werden. Hierfür ist eine sichere Verbindung mit ausreichender Bandbreite notwendig. Sichere Netzwerkverbindungen können z.B. mittels VPN hergestellt werden. Wie in Abbildung 3 gezeigt, wird dedizierte Hardware und das ssh-Protokoll zur verschlüsselten Datenübertragung verwendet. Ein solcher Fernzugriff ermöglicht es, diverse Aufgaben durchzuführen. Hierzu gehören die Softwareinstallation und die Analyse von System- und Anwendungsproblemen. Über entsprechende Komponenten können Serversysteme auch ein- und ausgeschaltet werden. Sonstige hardwarenahe Aufgaben müssen ggf. vor Ort durchgeführt werden.

Um den Zugriff durch einen Dienstleister einzuschränken, kann der Zugang auf definierte Systeme des Kunden limitiert werden. Vertragliche Regelungen und der ausschließliche Verbleib von Daten beim Kunden gewähren zusätzliche Sicherheit.

Service Level

Service-Level-Vereinbarungen definieren die Rahmenbedingungen des Supports. Dazu gehören z.B.:

- Erreichbarkeit
- Reaktionszeiten
- Fristen für Lösungsfindung und -umsetzung
- Leistungsumfänge
- Verfügbarkeit von Systemen und Anwendungen
- Länge und Frequenz von Vor-Ort-Einsätzen
- Konventionalstrafen
- Mitwirkung des Kunden

Im Rahmen des Servicekonzepts legen die Vertragspartner außerdem folgende Eckpunkte fest, die eine erfolgreiche Bearbeitung des Projekts ermöglichen:

- Verantwortliche
- Verantwortlichkeiten
- Ansprechpartner
- Berichterstattung
- Besprechungen

- Besprechungstermine
- Abnahmekriterien
- Eskalationswege

Die Kommunikation zwischen Kunde und Dienstleister erfolgt über Telefon, E-Mail, Fax und Online Supportformulare. Dabei sollte auf einen effizienten Einsatz geachtet werden. Der Kunde kann z.B. konkrete Ansprechpartner benennen, die die Kommunikation mit dem Dienstleister kanalisieren. Außerdem können die zu verwendenden Kommunikationssysteme für die erste Kontaktaufnahme vorgegeben werden, so dass z.B. Supportfälle immer per E-Mail gemeldet werden. Somit wird die Problemsituation in konkreter Form vom Betroffenen selber geschildert und kann auch ggf. automatisch im Trouble Ticket System aufgenommen werden.

Zusammenfassung

Oft übernehmen externe Dienstleister IT-Serviceleistungen für Produktentwicklung und Engineering. Die erforderlichen Leistungen können über geeignete Wartungsmethoden von entfernten Standorten aus durchgeführt werden. Service Level Vereinbarungen legen die Bedingungen und die Qualität der Leistungen fest. Überwachungsmechanismen und ein Trouble Ticket System sind wichtige Hilfsmittel für den CA-Systemservice.

Jan Martini, GNS Systems GmbH
jan.martini@gns-systems.de

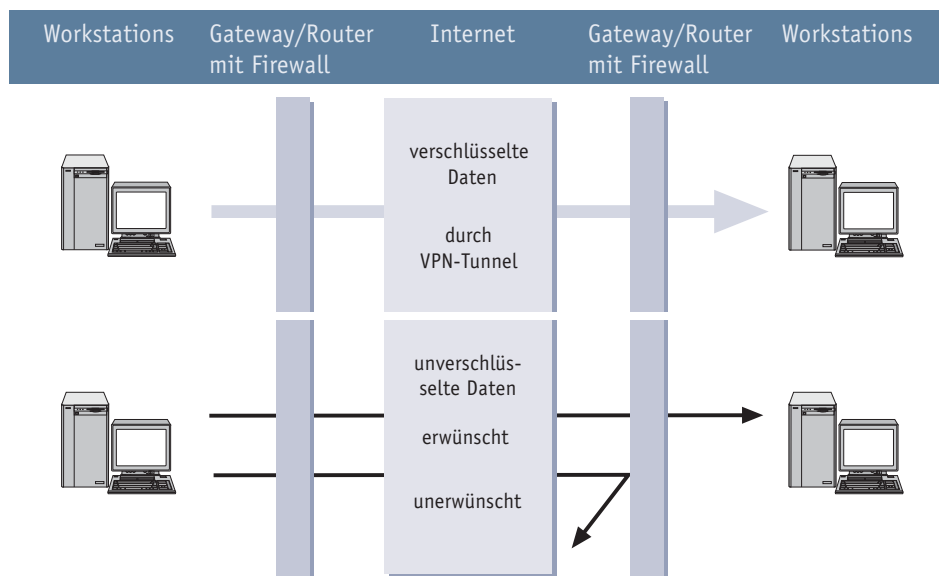


Abb. 3: Darstellung einer VPN-Verbindung

Termine

Im Augenblick stehen noch keine Veranstaltungen für 2007 fest.

Die nächsten Messen, Tagungen und Konferenzen teilen wir Ihnen in den nächsten Ausgaben oder auf unseren Webseiten mit.



GNS mbH

Am Gaußberg 2
38114 Braunschweig
Telefon: 05 31-8 01 12 0
Fax: 05 31-8 01 12 79
www.gns-mbh.com



GNS Systems GmbH

Am Gaußberg 2
38114 Braunschweig
Telefon: 05 31-12387 0
Fax: 05 31-12387 11
www.gns-systems.de

Impressum

Ausgabe 3/2006
Erscheinungstermin: Dezember 2006
Herausgeber: GNS Systems GmbH

Verantwortlich: Jan Martini
Redaktion: Anette Träger
Layout: Anette Träger

Alle Rechte vorbehalten.
Vervielfältigung, auch auszugsweise, nur mit schriftlicher Genehmigung des Herausgebers.

Videofunktionalität in Animator3

Der Import von Videos geschieht im "Video 0/1"-Dialog über "File->Open". Der meist in Versuchs-AVI-Dateien verwendete Codec IV50 kann von *Animator3* nicht direkt eingelesen werden. Trifft *Animator3* auf eine solche Datei, startet er das *mencoder*-Script im INSTALLATION/Utilities-Verzeichnis. Dieses erstellt mit Hilfe des Freeware-Programms *mencoder* eine Datei im lesbaren Format (default = mjpeg), welche dann eingelesen wird.

Im *mencoder*-Script können der Aufruf und die Konvertierungsoptionen angepasst werden. *mencoder* ist Teil des *Mplayer*-Pakets. Es wird nicht mit der *Animator3*-Distribution geliefert, sondern muss gesondert installiert werden.

Nachfolgend ein Beispiel zur Synchronisation von Video und Berechnung:

Die Berechnung wurde in Sekunden mit den Zeitschritten 0, 0.002, 0.004, 0.006 ... 0.1 durchgeführt. Das Video liegt in Millisekunden mit 500 Bildern von 0 bis

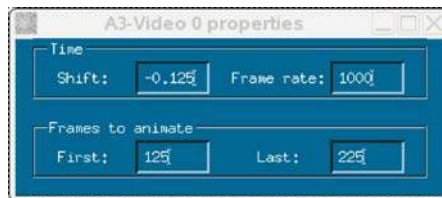


Abb. 1: Zeitsynchronisation im Video-Properties-Dialog

500 Millisekunden vor, wobei der Berechnungsnullpunkt 125ms im Video entspricht.

Die Synchronisation des Zeitsignals von Video und Berechnung geschieht über die Angabe des Zeitpunktes (Berechnungszeiteinheit) des ersten und des letzten Videobildes. Für obiges Beispiel gilt dann:

```
vi0 tip fir -0.125
vi0 tip las 0.375
```

Der "Video->Properties"-Dialog (Abb. 1) bietet die Möglichkeit, diese Information als "(time)shift" (-0.125) und "frame rate"

(1000) vorzugehen. Hier kann man auch die darzustellenden Bilder des Videos einschränken (frames to animate), um keinen Vor- bzw. Nachlauf zu haben. Für obiges Beispiel z.B.:

```
vi0 fra fir 125
vi0 fra las 225
```

Will man nun noch das Video im Hauptfenster zusammen mit der Simulation betrachten, aktiviert man den Schalter "Display->As background" im Videodialog.

Die im Hauptfenster dargestellte Animation kann als Video exportiert werden. Dies erfolgt mit nachfolgender Kommando-sequenz:

opt avc ...	Angabe des Komprimierungsverfahrens
xcm gwi/ ghe pixel	Festlegung der Größe des Grafikfensters
rec avi sta filename	Start der Videoaufnahme
ani fo1	Einmaliges Durchlaufen der Animation
rec avi sto	Beendigung der Videoaufnahme

So kann *Animator3* auch zur Videokonvertierung bzw. zum Videoschnitt benutzt werden.

Carsten Thunert, GNS mbH
animator@gns-mbh.com

