

РАСПРЕДЕЛЕННЫЕ ВЫЧИСЛЕНИЯ НА ПЛАТФОРМЕ .NET

Бурак А.А.

Кафедра информационных технологий автоматизированных систем

Научный руководитель: Герман О. В., к.т.н., доцент

e-mail: brain992@mail.ru

Аннотация – Распределенные вычисления – это способ решения трудоемких вычислительных задач с использованием нескольких компьютеров, чаще всего объединенных в параллельную вычислительную систему. Особенностью распределенных многопроцессорных вычислительных систем, в отличие от локальных суперкомпьютеров, является возможность неограниченного наращивания производительности за счет масштабирования.

Ключевые слова: распределенные вычисления, WCF, MPI.

Широкое распространение параллельных архитектур вычислительных систем вызывает повышение интереса к средствам разработки программного обеспечения, способного максимально полно использовать аппаратные ресурсы вычислительной системы. Однако к текущему моменту имеется определенный разрыв между имеющимися на потребительском рынке технологиями аппаратной реализации параллелизма и программными средствами их поддержки. Так, если многоядерные компьютеры общего назначения стали нормой в середине текущего десятилетия, то появление OpenMP — популярного стандарта разработки программ для подобных систем — отмечено почти десятью годами ранее. Практически в то же время возник и стандарт MPI, описывающий способы передачи сообщений между процессами в распределенной среде.

Развитие обоих данных стандартов, выражающееся только в расширении функциональности без адаптации парадигм к объектно-ориентированному подходу, приводит к тому, что они оказываются несовместимы с современными платформами программирования, такими как Microsoft .NET Framework. Поэтому разработчикам этих платформ приходится прилагать дополнительные усилия по внедрению средств параллелизма в свои продукты.

Тем не менее, начиная с третьей версии, .NET Framework включает в себя WindowsCommunicationFoundation (WCF) — унифицированную технологию создания всех видов распределенных приложений на платформе Microsoft. К сожалению, данная технология зачастую понимается только как каркас для работы с Web-службами на основе XML, что напрасно и мешает рассматривать WCF как эффективное средство для организации параллельных вычислений.

Самый простой тест, призванный показать эффективность обмена данными между компонентами вычислительной сети, заключается в отправке массива вещественных чисел двойной точности с одного узла на другой и обратно с фиксацией времени,

затраченного на выполнение данных операций. Выполним данную задачу стандартом MPI (алгоритм на языке C++) и технологией WCF (на языке C#).

Программный код WCF обладает несколькими важными достоинствами относительно MPI, а именно: единообразное и однократное описание форматов передаваемых данных, реализуемое в интерфейсе (контракте); простая реализация вызовов удаленных методов; использование объектно-ориентированного подхода к разработке.

Таким образом, можно утверждать, что применение WCF существенно облегчает задачу написания распределенных приложений.

Табл. 1. Результаты теста эффективности обмена данными

Размер массива	Время обмена данных, мс		Преимущество MPI, %
	MPI	WCF	
2	0,18	0,66	266,67
4	0,18	0,68	277,78
32	0,2	0,81	305,00
128	0,38	1,06	178,95
256	0,506	1,31	158,89
512	0,91	1,66	82,42
1024	1,528	2,58	68,85

Из данной таблицы следует, что, к сожалению, технология WCF не пригодна для разработки программ, требующих частого межпроцессного обмена данными малого объема. Однако при построении распределенных вычислительных приложений с малым числом межпроцессных коммуникаций показывает хорошие результаты. Технология WCF, предназначенная для построения приложений такого рода, обеспечивает гораздо более простой способ межпроцессной коммуникации, нежели реализация приложений в MPI.

[1] OpenMP Reference. OpenMP Architecture Review Board, 2008 г.

[2] MPI 2.1 Reference. University of Tennessee, 2008г.

[3] Параллельное программирование в .NET. Тихонов, И. В. Иркутск, 2009. Труды XIV Байкальской Всероссийской конференции «Информационные и математические технологии в науке и управлении».