Cloud Task Parallelization in .NET (Client Components)

Version: 1.0.0.0

Project Website

http://concurrency.ch/Projects/TaskParallelism

Prerequisites

Microsoft .NET Framework 4.5

Installation

The DLLs in the folder “Library” can be directly used or deployed as part of an application.

Configuration

The client requires a remote task parallelization web service. The following information needs to be configured at the client:

- URL of the task parallelization web service
- Authorization token (string) that is registered at the web service and authorizes the client to use the service.

Public Parallelization Service

Our preinstalled public task parallelization service which is available for free. This service offers 12 cores for evaluation purposes; more resources may be provided on request.

- URL: http://tasks.concurrency.ch
- Request your individual authorization token via http://concurrency.ch/Contact form.

Custom Parallelization Service

Alternatively, you can set up your own task parallelization web service (e.g. via HTTPS) that uses a MS HPC cluster (download server components from the project website).

Usage

It is sufficient to reference the library `HSR.CloudTaskParallelism.Client.Runtime` in the .NET client program. Distributed tasks can be run (started and awaited) over an instance of the `Distribution` class, as outlined below. The URL of the task parallelization service and the corresponding authorization token need be specified on construction of the `Distribution` class.

```
public void FactorizeNumberSetRemotely(long[] inputs) {
  var distribution = new Distribution("http://tasks.concurrency.ch ", "my_token");
  var taskList = new List<DistributedTask<long>>();
  foreach (long number in inputs) {
  var task = DistributedTask.New(() => _Factorize(number));
    taskList.Add(task);
  }
  distribution.Start(taskList);
```

```
  foreach (var task in taskList) {
    Console.WriteLine(task.Result);
  }
}

// TASK CODE
private long _Factorize(long number) {
  for (long k = 2; k <= Math.Sqrt(number); k++) {
    if (number % k == 0) { return k; }
  }
  return number; // not factorizable
}
```

Samples

A series of sample applications are provided as Visual Studio 2012 projects. In the application settings (`AppSettings.settings`), the URL and authorization token of the task parallelization service need to be configured.

Restrictions

The following restrictions apply with regard to the code that is executed by distributed tasks

- No nested tasks
- No call of libraries
- No unmanaged code (security reasons)

The current version does not yet support the following language features to be used for distributed tasks:

- Type polymorphism by inheritance, interfaces, or delegates
- Struct and, in particular, nullable types
- Multidimensional arrays (nested/jagged arrays are supported)
- Ref and out parameters
- Exceptions

License