

\$2 an hour workers join virtual production line

April 19, 2012

by John Lister

Manufacturers already break down production into small tasks and outsource them to the developing world. Now a company thinks it can do the same with computer-related services.

CloudFactory allows companies in prosperous nations to outsource relatively unskilled work to people in developing nations, benefiting from the much lower wages demanded in those countries. That's nothing new in itself, but the CloudFactory idea is to adopt the assembly line approach.



It works by allowing companies to take the tasks they need and break them down into individual parts. It says this could cover any task that doesn't need specialized and highly-trained workers, but does still need humans rather than computers.

Once the tasks are broken down, CloudFactory sends the work out to available staff around the world, the idea being to make the best use of availability and timezones at any specific moment. As with manufacturing, the theory is that if particular staff are each working on a very specific task, they'll be able to get to grips with it more quickly, get into a routine and maximize output.

The workers concerned won't be getting rich, or at least not by Western standards. Though jobs are custom priced, it appears the [basic set-up is \\$2 an hour for each worker](#) with an additional \$20 per hour fee for the entire project going to CloudFactory.

For tasks that can be carried out automatically, the company offers a range of "robots" that can be plugged into the virtual production line. These carry out tasks such as media file conversion, keyword matching and content scraping. The fee for the robots is on a per task basis, so it's hard to tell if they could wind up "earning" a higher fee than the workers.

The company, based in Nepal, says it aims to [employ a million workers within five years](#). It insists that not only does the money it pay allow people to "make a living" but that it will make sure they receive education and training.