ABSTRACT

In real-time middleware, latency is a critical aspect. When the input rate exceeds a certain threshold, queuing will result in an exponentially increasing delay. Distributed computing enables scaling so that this growing latency is kept at a constant minimum. When developing a generic framework, redundancy, platform independence, fault tolerance and transparency are important features that need to be taken care of. The authors’ test case, a localization system that is used to track multiple objects, is an example of such a heavy loaded and latency critical system. This specific test case requires that the framework quickly scales and reacts to changing loads, making this research challenging and innovative.

Keywords: Computer Science, Cross-Platform, Distributed, Fault Tolerance, Load Balancing, Transparent

INTRODUCTION

Opportunistic Seamless Localization (OSL) (Weyn, 2011) is a localization system (Porretta et al. 2008) that merges several possible localization technologies such as Wi-Fi, GPS, GSM, Bluetooth, Wireless Sensor Networks or inertial sensors.

As Figure 1 shows, OSL consists of multiple logical blocks of which the localization engine is the most important. This engine receives the raw measurement data coming from the different devices using different localization technologies and calculates the most likely position of an object. It is the most computational intensive component of OSL.

DOI: 10.4018/ijaras.2013100104
A one-tier implementation is able to determine the positions using Wi-Fi measurements of approximately 8 objects per second, when using fusion states of 250 particles, without any noticeable delay. Such result was reached on a standard laptop, using Windows and Visual Studio 2010, with a quad-core 2.0 GHz processor and 4GB DDR3 RAM. Eventually, OSL will be used to track several thousands of objects which would result in an exponentially increasing queuing delay. Since this localization system will be used to track all these objects in real-time, the framework is built to minimize any queuing time. Magalhães et al. (1993) define soft and hard deadlines in a real-time system. Meeting a hard deadline is critical for the system while missing a soft deadline decreases the performance but is not critical. In order to enable OSL to track several thousands of objects, it will be distributed. Because the calculated results need to be useful for real-time tracking, a soft deadline of 1 second is provided. Within this timeframe an estimated position should be returned.

Vertical scaling (running OSL on a server with better specifications) is one possible solution, but it has its limitations and is very expensive. On top of this, processing the data requires some time and it is not possible to eliminate this processing time by adding extra hardware. Another solution is distributing the software across a network of computers. Michael et al. (2007) concluded that horizontal scaling has improved performance and a better price/performance ratio. This conclusion made us decide to scale OSL horizontally.

The implementation and research concerning this horizontal scaling can be summarized in the following hypothesis:

A redundant, fault tolerant, platform independent and real-time framework can be developed to enable scalable and distributed computing.

This hypothesis can be divided into the following claims:

1. Distribution and scaling can be autonomous and self-arranging.
2. A redundant, fault tolerant and platform independent framework can be developed requiring a minimum of modifications by the programmer and transparent to the end-user.
3. Each individual engine can receive tasks independently of other servers so that each server is optimally used.

In OSL, shared data is required. This shared data consists of multiple variable states that should be available for all engines. Besides this shared data, it is computational intensive because of the huge amount of different iterative calculations. This is in contrast to an application where one very computing intensive calculation is divided. The calculations that are performed by OSL consist, among others, of a Bayesian filtering, a dynamic motion model...
A Knowledge-Based System for Sharing and Reusing Tacit Knowledge in Robotic Manufacturing
Lei Wang, Yajie Tian, Tetsuo Sawaragi and Yukio Horiguchi (2010). *International Journal of Knowledge and Systems Science* (pp. 61-78).
www.igi-global.com/article/knowledge-based-system-sharing-reusing/48845?camid=4v1a