Test effettuati ed appunti relativi

Si evita di riportare i test effettuati al termine del mese di gennaio e nei primi giorni di febbraio in quanto essi, allo stato dell'arte, non sono più molto significativi. Basti sapere che in buona parte sono stati compiuti per identificare quale fosse la condizione di locking preferibile per le diverse piattaforme.
Nella fattispecie questo ha condotto ad identificare l'uso delle seguenti API:

- WaitForSingleObject per la sincronizzazione delle header dei files in Windows.
- Boost Mutex per la sincronizzazione delle header dei files in piattaforme Unix/LInux.

Queste due soluzioni consentono di ridurre sensibilmente il tempo della *Critical Section* riducendo ai minimi termini il tempo inutilmente perso nell'attesa che la risorsa voluta risulti disponibile. Si noti che in Windows, inoltre, si ha una ulteriore garanzia, ovvero che le richieste sulla risorsa in esame, poste opportunamente in coda, vengano evase nell'ordine in cui sono state appunto richieste, con un beneficio evidente sul tempo medio d'attesa delle singole operazioni.

Ulteriori test ci hanno consentito di isolare le parti più rappresentative della *Critical Section* per escludere dalla stessa quelle parti che avevano un impatto inatteso. Quanto detto ha condotto alla revisione del sistema di *Last Recently Used* applicato sul B+Tree, ne ha evidenziato una schiacciante superiorità rispetto al BTree standard adottato da 3D da quasi un ventennio ed ha posto luce sul fatto che le procedure di logging richiedono un tempo sorprendentemente elevato per essere condotte a termine.
Questo ha in fine condotto alla realizzazione di procedure a thread anche presso i server Slave, procedure che tramite una *Lockless Queue* consentono di parallelizzare operazioni in fine onerose come la stesura del file di log standard del server, “necessario come il pane”.
Tali interventi troveranno presto un riflesso diretto sulla stesura del file di Registro (xreg) che è stato intenzionalmente escluso da questi test.

Veniamo quindi al tentativo di utilizzo del BuddyXML: Esso ha dato prime dimostrazioni di essere piuttosto rapido e performante ma in quella fase i test si sono dimostrati sorprendentemente variabili nei risultati dandoci la netta sensazione che non fossero di fatto molto significativi.
Lo sviluppo inerente verrà quindi rimandato ad una sessione successiva ed il risultato sino ad ora ottenuto viene “tenuto in caldo”.

I test sul BuddyXML sono stati però l'occasione per verificare anche ulteriori problemi. Il *locking* dei singoli record (del file UDP ma anche del file pilota dei titoli) ha un consumo di tempo sensibilmente superiore alle attese ed in buona parte dei casi (specialmente in inserimento) può e deve essere evitato. Parimenti il documento che viene scritto (inserito o modificato poco cambia) viene riletto dopo esser stato scritto con evidenti conseguenze sulle performance. Anche questo può e deve essere cambiato.